Modeling molecular signaling and gene expression using Dynamic Nested Effects Models

DISSERTATION ZUR ERLANGUNG DES DOKTORGRADES DER
NATURWISSENSCHAFTEN(DR.RER.NAT.) DER FAKULTÄT FÜR BIOLOGIE
UND VORKLINISCHE MEDIZIN DER UNIVERSITÄT REGENSBURG

vorgelegt von
Benedict Nchang Anchang

aus

Yaounde, Kamerun

im Jahr
2011

Modeling molecular signaling and gene expression using Dynamic Nested Effects Models

Benedict Nchang Anchang

Department of Statistical Bioinformatics

University of Regensburg

A thesis submitted for the degree of

Doctor of Natural Sciences (Dr.rer.nat.) in the Faculty of Biology and Preclinical Medicine.

October 2011

1. Supervisor: Prof. Dr. Rainer Spang

2. Supervisor: Prof. Dr. Wolfram Gronwald

Date of application for admission: 02-11-2011

Day of the defense: 09-12-2011

Signature from Author:

Signature from head of PhD committee:

Abstract

Cellular decision making in differentiation, proliferation or apoptosis is mediated by molecular signaling processes, which control the regulation and expression of genes. *Vice versa*, the expression of genes can trigger the activity of signaling pathways. I summarize methodology by Markowetz *et al.* known as the Nested Effects Models (NEMs) to reconstruct static non-transcriptional networks using subset relationships from perturbation data and bring out its limitation to model slow-going biological processes like cell differentiation. I introduce and describe new statistical methodologies called Dynamic Nested Effects Models (DNEMs) and Cyclic Dynamic Nested Effects Models (CDNEMs) for analyzing the temporal interplay of cell signaling and gene expression. DNEMs and CDNEMs are Bayesian models of signal propagation in a network. They decompose observed time delays of multiple step signaling processes into single steps. Time delays are assumed to be exponentially distributed. Rate constants of signal propagation are model parameters, whose joint posterior distribution is assessed *via* Gibbs sampling. They hold information on the interplay of different forms of biological signal propagation: Molecular signaling in the cytoplasm acts at high rates, direct signal propagation *via* transcription and translation at intermediate rates, while secondary effects operate at low rates. I evaluate my methods in simulation experiments and demonstrate their practical applications to embryonic stem cell development in mice. The results from these models explain how stem cells could succeed to carry out differentiation to specialized cells of the body such as muscle cells or neurons, a process that goes in one direction. The inferred molecular communication underlying such a process proposes how organisms protect themselves against the reversal of cell differentiation and thereby against cancer.

Zusammenfassung

Die zelluläre Entscheidungsfindung in der Differenzierung, der Zellproliferation oder der Apoptose wird durch molekulare Signalprozesse, die die Genregulation und -expression steuern, vermittelt. Andersherum kann die Genexpression die Aktivität der Signalverläufe auslösen. Ich fasse die Methodik von Markowetz *et al.*, bekannt als Nested Effects Models (NEMs), zusammen um statische nicht-transkriptionelle Netzwerke anhand von Teilmengenbeziehungen aus Perturbationsdaten zu rekonstruieren. Dabei zeige ich die Anwendungsgrenzen dieser Methodik zur Modellierung langsamlaufender biologischer Prozesse wie z.B. Zelldifferenzierung. In dieser Arbeit führe ich neue statistische Methoden namens "Dynamic Nested Effects Models" (DNEMs) und "Cyclic Dynamic Nested Effects Models" (CDNEMs) mitsamt deren Beschreibung für die Analyse des zeitlichen Zusammenspiels von zellulärer Signalübertragung und Genexpression ein. DNEMs und CDNEMs sind Bayessche Modelle der Signalweiterleitung in einem Netzwerk. Sie zerlegen beobachtete Zeitverzögerungen der Signalprozesse von mehreren Schritten in einzelne Schritte. Zeitverzögerungen werden als exponential verteilt angenommen. Geschwindigkeitskonstanten der Signalweiterleitung sind Modellparameter, deren gemeinsame posteriori-Verteilung über Gibbs-Sampling beurteilt wird. Sie enthalten Informationen über das Zusammenspiel der verschiedenen Arten von biologischer Signalweiterleitung: Molekulare Signalweiterleitung ins Zytoplasma findet mit hoher, direkte Signalweiterleitung über Transkription und Translation mit mittlerer und sekundaere Effekte mit niedriger Geschwindigkeit statt. Ich beurteile meine Methoden mit numerischen Simulationsexperimenten und zeige ihre praktische Anwendbarkeit anhand von Daten aus murinen embryonischen Stammzellen. Die Ergebnisse aus diesen Modellen erläutern wie es Stammzellen gelingt zu spezialisierten Zellen des Körpers wie Muskelzellen oder Nervenzellen zu differenzieren. Der Prozess im wesentlichen in eine Richtung. Die hieraus abgeleiteten molekularen Kommunikationsmechanismen eines solchen Prozesses stellen dar, wie sich ein Organismus vor der Umkehrung der Zelldifferenzierung und damit vor Krebs schützen kann.

To my Father and late Mother

Acknowledgements

This work was carried out at the Institute of Functional Genomics and Bioinformatics in the department of Statistical Bioinformatics, all part of the University of Regensburg. It was supported by the Bavarian Genome Network BayGene and the ReForM-M program of the Regensburg School of Medicine. I would like to thank all past and present colleagues for their moral, friendly and technical support.

Special thanks goes to my supervisor *Rainer Spang* for suggesting the topic, his scientific support, and the opportunity to write this thesis under his guidance. Further thanks goes to *Wolfram Gronwald* and *Peter Oefner* for their supervision and counsel during my PhD work.

The successful completion of this thesis is also the result of scientific advice and recommendations of other collaborators. I am particularly grateful to *Achim Tresch*, *Florian Markowetz*, and *Marcel O Vlad* who is of late.

Final gratitude goes to the moral support of my family and friends especially Kilian Awemo who did not fully understand my research topic yet went through the pains of reading my work and offering a few recommendations.

Publications Part of this thesis has been published in peer-reviewed journals. The main content of Chapter 4 has been published in the Proceedings of National Academy of Science(PNAS) and has been presented at the International Conference on Intelligent Systems for Molecular Biology(ISMB), Boston, USA, 9-13 July, 2010. ISMB is the leading Bioinformatics conference world wide. The program committee has selected the paper for their highlight track which consists of the most significant Bioinformatics contributions of the year.

Figures This thesis reproduces/adapts figures from other publications. Figures 1.1 and 1.3 from chapter 1 are adapted with permission from Alberts *et al.*(2008) (1). Figure 1.4 has been taken from Wikipedia.org, where it is published under GNU Free Documentation Licence. Figure 2.5 is reproduced from Vaske *et al.*(2009) (2). Figure 1.7 is reproduced from Alon(2007) (3) and finally the adapted Figures 1.2 and 2.1 are from Niwa(2007) (4) and Wagner 2001 respectively.

Contents

List of Figures xi

List of Tables xiii

Glossary xv

1 Introduction 1

1.1 Cell decision making in biological processes 1

1.1.1 Key players in the molecular mechanism in early stem cell differentiation in mouse 3

1.2 Properties of cellular decision making processes 5

1.3 Cellular decision making processes can be represented by hierarchies . . 6

1.4 Statistical methods for analyzing decision making processes 8

1.4.1 Learning from experimental interventions 8

1.5 Motivation for dynamic modeling of cell decision making processes . . . 9

1.5.1 The Feed-Forward Loop Network motif 9

1.5.2 Feed-Back Loop Network motif 10

1.6 Thesis Organization 12

1.6.1 Nested Effects Models 12

1.6.2 Gibbs sampling and Nested Effects Models 12

1.6.3 Dynamic Nested Effects Models 12

1.6.4 Cyclic Dynamic Nested Effects Models 13

2 Nested Effects Models 15

2.1 Nested Effects Models(NEMs) 16

2.1.1 Marginal likelihood scoring 18

2.1.2 Maximum likelihood scoring 20
2.1.3 NEMs as a Bayesian network 21
2.1.4 Factor graph NEMs 23
2.1.4.1 Structure of factor graph NEMs and Network inference 24
2.2 Network learning algorithms in NEMs 27
2.2.1 Pairwise and triple search 27
2.2.2 Module networks 28
2.2.3 Greedy hill search 28
2.3 Overview and differences on Nested effects models 28

3 Gibbs sampling 31

3.1 Background on Gibbs sampling 31
3.2 Gibbs sampling 32
3.2.1 The Markov chain property 33
3.2.2 The Monte Carlo property 34
3.2.3 Variations of Gibbs sampling 34
3.2.3.1 Blocked Gibbs sampler 35
3.2.3.2 Collapsed Gibbs sampler 35
3.2.3.3 Gibbs sampler with ordered overrelaxation 35
3.2.4 Extensions of Gibbs sampling 35
3.2.5 Assessing convergence 36
3.2.6 Monitoring the convergence of each parameter of interest 36
3.2.6.1 Trace plots 37
3.2.6.2 Autocorrelation 37
3.2.6.3 Gelman and Rubin diagnostic 37
3.2.6.4 Geweke diagnostic 38
3.2.6.5 Raftery and Lewis diagnostic 38
3.2.6.6 Heidelberg and Welch diagnostic 39
3.3 Motivation of Gibbs sampling in modeling the dynamics of a cell decision process 39

4 Dynamic Nested Effects Models 41

4.1 Dynamic Nested Effects Models(DNEMs) 41

4.1.1 DNEM algorithm 44

4.1.1.1 Model parameters 44

4.1.1.2 Prior distributions for model parameters 45

4.1.1.3 Probability density of signal propagation between input and output signal 46

4.1.1.4 Probability density of signal times generalized to phase-type distributions 47

4.1.1.5 Probability density function when alternative paths do not share edges 48

4.1.1.6 Probability density function when alternative paths share edges 49

4.1.1.7 Sensitivity analysis between independent and dependent signals 50

4.1.1.8 Marginal likelihood for discrete models 50

4.1.1.9 Marginal likelihood for continuous models 52

4.1.1.10 Discrete Gibbs sampling 52

4.1.1.11 Inference of E-gene positions 53

4.1.1.12 Inference of signal flow 54

4.1.1.13 Model comparisons using DNEMs 55

4.1.2 Speed up by stochasticity effects in signaling networks 56

4.2 Simulation results 58

4.2.1 Data generation 58

4.2.2 A sparse network 58

4.2.3 Dense networks and detection of shortcut pathways 61

4.3 Application to cell differentiation in embryonic stem cells 61

4.3.1 Data preprocessing 62

4.3.2 Binary data 62

4.3.3 Time series analysis 63

4.3.4 Stability analysis 63

4.3.5 Stability of network topologies in the static NEM analysis 64

4.3.6 Stability of feed-forward loop detection in the DNEM analysis . 65

4.3.7 Decision between *Oct4* and *Tcl1* direction in network 65
4.3.8 Convergence and Mixing analysis of the Gibbs sampler 67
4.3.9 Inference of signaling in Network 67

5 Cyclic Dynamic Nested Effects Models(CDNEMs) 71
5.1 Model parameterization of CDNEMs 71
5.2 Probability density of signal propagation in a directed cyclic graph . . . 72
5.2.1 Algorithm to enumerate all paths in a directed cyclic graph between two nodes 72
5.2.2 Probability density of signal propagation in CDNEMs 73
5.3 Simulation results 75
5.3.1 Data generation 75
5.3.2 Results 75
5.3.2.1 Fully connected network with equal time delays 77
5.3.2.2 Cyclic dense network with varying time delays 77
5.3.2.3 Directed acyclic networks 79
5.3.2.4 Network with only one cycle 79
5.4 Application of CDNEMs to cell differentiation in embryonic stem cells . 79
5.4.1 Inference of signaling in Network during early ESC differentiation 80

6 Impact of Dynamic Nested Effects Models 85
6.1 Fast Cyclic Dynamic Nested Effects Models(FCDNEMs) 85
6.1.1 Model parameters for FCDNEMs 86
6.1.2 Marginal likelihood for discrete model 86
6.1.3 Using Priors for network structures and time delays 88
6.1.4 Network Learning for FCDNEMs 89
6.2 Application of FCDNEMs to cell differentiation in embryonic stem cells 90

7 Summary and Outlook 93
7.1 Conclusions 93
7.2 Future directions 94
7.2.1 Combinatorial perturbations 95
7.2.2 NEMs and drug interventions 95
7.3 Food for thought 96

8 Appendix **97**

8.1 Derivation of the Probability density of signal propagation along a linear path 97

References **99**

List of Figures

1.1 Cell decision making processes 2
1.2 Pluripotent lineages in the mouse embryo with key Transcription factors 5
1.3 Properties of cell decision making 6
1.4 Properties of cell decision making 7
1.5 cellular decision processes can be modeled as hierarchies 7
1.6 Transitive edge and Feed-Forward Loop 10
1.7 Feed-Forward Loops(FFLs) 11

2.1 Cellular networks underlying observable phenotypes 16
2.2 Cellular pathways can be reconstructed from the nested structure of downstream effects 17
2.3 Bayesian Nested effects models 22
2.4 Bayesian network next to corresponding factor graph 25
2.5 Structure of factor graph for network inference in Factor graph NEMs . 26
2.6 A guide to the literature on NEMs 30

4.1 Time delays in a given network 42
4.2 Idea of DNEMs in an elementary example 43
4.3 Parameterization of DNEM. 45
4.4 Graph of 5 S-genes with alternative paths sharing an edge. 49
4.5 QQ-plots comparing distributions of signal times between two nodes S_1 and S_5 in the graph in Figure 4.4 under dependent and independent assumptions. 51
4.6 Updating E-gene positions inside Gibbs. 54
4.7 Speed up by stochasticity effect of DNEMs 57

4.8 The transitive network used for simulation 59
4.9 Heat map of the posterior distribution of average time delays 60
4.10 Stem cell data analysis 64
4.11 Stability Analysis 66
4.12 Diagnostic Plots for the Gibbs sampler 68
4.13 DNEM inference on signal propagation 69

5.1 A directed cyclic graph 72
5.2 A directed cyclic graph with parameters 76
5.3 Cyclic DNEM models with 3 nodes and varying rates 77
5.4 Heat map of the posterior distribution of CDNEM model 78
5.5 Fully connected directed input graph with 6 key TFs 80
5.6 Heatmap of posterior distribution and output graph of signal flow under non-stochastic assumptions 81
5.7 Heatmap of posterior distribution with output graph of signal flow under stochastic signaling 82

6.1 Standard NEM with 3 nodes 87
6.2 Inferred network for murine stem cell development 91

List of Tables

2.1 The distribution of binary data 18

5.1 Algorithm to enumerate all paths between two nodes in a graph 73
5.2 Simulated time delays for edges in cyclic graph 76

Glossary

α	Probability to observe a false positive effect
β	Probability to observe a false negative effect
κ	Discretization threshold parameter
$\mathcal{E}_i$	The set of E-genes attached to the same S-gene S_i
Θ	Parameter for E-gene positions
$\theta_k = i$	Position parameter for E-gene E_k linked to S_i
D	Perturbation data matrix
D_{ikls}	Observed E-gene expression level E_k after perturbation S_i for replicate experiment l in time point t_s. Sometimes also represented as e_{ikls}
E	Effect genes known as E-genes. Also known as observables O
$H^{'}$	Set of Hidden nodes in a graph
K	Set of rate parameters between S-S-genes and S-E-genes
S	Signaling genes known as S-genes
BIC	Bayesian information criterion; a criterion for model selection among a class of parametric models with different numbers of parameters
CDNEMs	Cyclic Dynamic Nested Effects Models
DAG	Directed acyclic graph
DCG	Directed cyclic graph
DIC	Deviance Information Criteria ; a criterion for model selection particularly useful in Bayesian model selection problems where the posterior distributions of the models have been obtained by Markov chain Monte Carlo (MCMC) simulation
DNEMs	Dynamic Nested Effects Models
EM	Expectation-Maximization; an algorithm for solving incomplete data problems
ESC	Embryonic stem cells
FBLs	Feed-Back Loops

GLOSSARY

FCDNEMs	Fast Cyclic Dynamic Nested Effects Models
FFLs	Feed-Forward Loops
HMG	High-Mobility Group; a group of chromosomal proteins that help with transcription, replication, recombination, and DNA repair
ICM	Inner cell mass
LIF	Leukemia Inhibitory Factor; a cytokine that affects cell growth and development
MAP	Maximum a posteriori probability. The MAP estimate is a mode of the posterior distribution
MAP kinase	Mitogen-activated protein kinase; a serine/threonine-specific protein kinase that responds to extracellular stimuli (mitogens, osmotic stress, heat shock and proinflammatory cytokines) and regulate various cellular activities, such as gene expression, mitosis, differentiation, proliferation, and cell survival/apoptosis
MCMC	Markov chain Monte Carlo; a class of algorithms for sampling from probability distributions based on constructing a Markov chain that has the desired distribution as its equilibrium distribution
NEMs	Nested Effects Models; a class of models for the analysis of non-transcriptional signalling networks. NEMs infer the graph of upstream/downstream relations for a set of signalling genes from perturbation effects.
PH	The phase-type distribution of the time until an absortion state in a Markov chain
RA	Retinoic Acid; is a metabolite of vitamin A (retinol) that mediates the functions of vitamin A required for growth and development
RNA	Ribonucleic acid
RNAi	RNA interference
TE	Trophectoderm
TFs	Transcription factors

1

Introduction

Cellular decision making in biological processes such as differentiation, proliferation or cell death is mediated by molecular signaling processes, which control the regulation and expression of genes. Changes in gene expression can activate further signaling processes, leading to secondary effects, which themselves give rise to tertiary effects and so on. The result is an intricate interplay of cell signaling and gene regulation. While protein modification in the cytoplasm can propagate signals in seconds, transcription and translation processes last hours, and secondary effects often become visible only after days. I develop statistical methodology that models the processes of cellular decision making using data, which reports downstream effects of molecular perturbations. In addition, I discuss which role such a model can play in biology and biomedicine. The first chapter gives some background on cell decision making processes and introduces key molecular players involved in stem cell differentiation. I focus on those properties which make it possible to systematically analyze and model such a process using high throughput perturbation data. I go further and discuss the methodology available for analyzing perturbation data with particular emphasis on limitation to time series data and provide motivation for taking the time into account.

.

1.1 Cell decision making in biological processes

Cellular decision making is involved in several biological processes such as cell division, cell proliferation, apoptosis or cell differentiation (Figure 1.1). Each of these processes

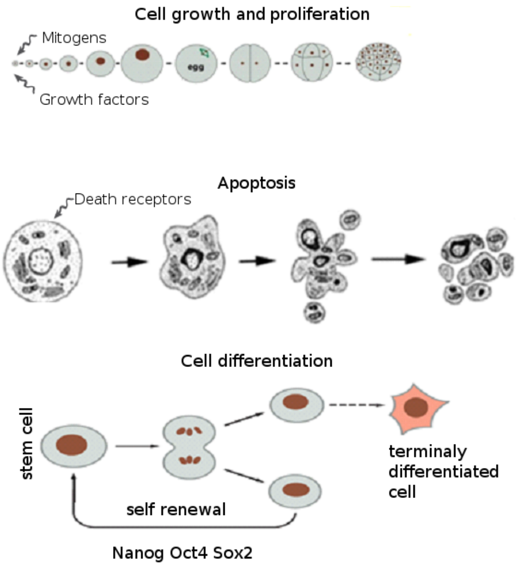


Figure 1.1: Cell decision making processes - Cell decision making is involved in cell growth, cell proliferation, apoptosis, and differentiation. Mitogens and growth factors induce the process of cell growth and cell division respectively. Both actions usually occur simultaneously. Death receptors trigger extrinsic apoptosis. The entire programming of a single death cell involves cell shrinkage, cell membrane blebbing, nuclear collapse, and apoptotic body formation. Stem cell development is mediated by both self renewal and differentiation. *Nanog*, *Sox2*, and *Oct4* play a key role in driving stem cells from a self renewal state into early differentiation. The figure is a modification from figures in (1).

is tightly regulated and controlled both by intracellular programs and extracellular signaling molecules whose mechanisms are still not clear. For example, the process of cell growth and cell division is stimulated by chemical substances like mitogens and growth factors respectively (1). Mitogens interact with cell surface receptors to trigger multiple intracellular signaling pathways during cell division. Although extensive research has been carried out in this area (1), its still not clear how a proliferating cell coordinates its growth with cell division so as to maintain its appropriate size. Alternatively, a process such as apoptosis which is useful for the elimination of unwanted cells in the body is triggered by death receptors on their surface. For example, *Fas* ligand, a transmembrane protein on the surface of a killer lymphocyte binds to the Fas receptor on the cell surface to trigger the extrinsic apoptosis pathway (5). Apoptosis can also be triggered from within the cell (6) and in some cases a combination of both external and internal signaling are involved to amplify the process (1). The mechanism underlying such a coordination is still not fully understood. In addition during the process of cell differentiation, stem cells need to decide when and how to move from the state of self renewal into differentiation. Such a complex process is governed by transcription networks known as developmental transcription networks(7), which need to make irreversible decisions on a slow time scale of one or more cell generations.

1.1.1 Key players in the molecular mechanism in early stem cell differentiation in mouse

The zygote can give rise to a complex organism through cell division, growth(proliferation) and cell specialization(differentiation). The first differentiation event, is the segregation of the trophectoderm (TE) and the inner cell mass (ICM) in the blastocyst. See the adapted Figure 1.2 from Niwa(2007) (4). The zygote is totipotent, developing into not only the fetus but also the placenta. The totipotency is maintained in cells known as blastomeres of the two-cell stage embryo. After mechanical separation of the blastomeres of the two-cell stage embryo, each blastomere is able to give rise to an adult organism, for example a mouse(8). These cells which have the ability to self-renew as well as differentiate into different cell types of the vertebrate embryo leading to the formation of an entire organism are known as embryonic stem cells(ESC), and the cells of the embryonic inner cell mass from which mouse ESC are derived are called pluripotent because of their ability to give rise to all of the cells of an embryo and adult(9). Pluripotency is maintained during ESC self-renewal through the prevention of differentiation and the promotion of proliferation. In fact, ESC can self-renew continuously for years if they are cultured under conditions that prevent their differentiation; for example, in the presence of leukemia inhibitory factor (LIF), a growth factor that is necessary for maintaining mouse ESC in a proliferative, undifferentiated state (10). Studies over the past few years have revealed the role that transcription factor networks play in the maintenance of ESC pluripotency (11, 12, 13, 14, 15, 16, 17, 18). From these studies we have transcription factors(TFs) that are pivotal for maintaining ESC in their self-renewal state when overexpressed such as *Nanog*, a homeobox transcription factor expressed throughout the pluripotent cells of the ICM with the particular goal to prevent endoderm differentiation (13, 19); *Oct3/4*, also called *Pou5f1*, an important regulator of pluripotency that acts as a gatekeeper to prevent ESC differentiation(16); and *Sox2*, a member of the Sox (SRY-related HMG box) family of proteins that bind to DNA through the 79-amino acid HMG(high mobility group) domain. *Sox2* is co-expressed with *Oct4* in the ICM (20). These TFs form a core transcriptional network associated with pluripotency in ESC (15, 18, 21, 22). Alternatively, the differentiation of mouse ESC can be induced by the expression of certain transcription factors. For example, the expression of the transcription factor *Gata6* in ESC results in their differentiation into primitive endoderm(12). Likewise, the expression of the caudal-type homeobox transcription factor 2 (*Cdx2*) induces ESC to differentiate into trophectoderm (23). Model relative to roles played by the above transcription factors during early embryonic development is shown in Figure 1.2. We have two types of transcription factors which play a role in ESC. (*i*) TFs with target genes that are expressed in undifferentiated ESC. (*ii*) TFs with target genes that are not expressed in undifferentiated cells but induced in differentiated ESC. The overexpression of type (*i*) TFs will maintain ESC in their self renewal state while overexpression of type (*ii*) will likely trigger the differentiation of ESC. These transcription factors function in combination with other processes and on the accessibility of their target genes, which are made more or less accessible by the modification of their DNA, histones, or chromatin structure. The challenge would be to understand how these TFs interact dynamically with each other to regulate the processes between self-renewal and differentiation. More so, understanding the mechanisms underlying the processes of pluripotency, self-renewal and subsequent differentiation in embryonic stem cells is central to utilizing them therapeutically.

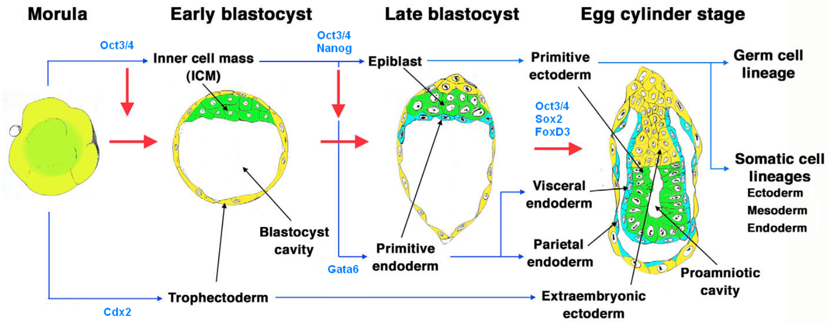


Figure 1.2: Pluripotent lineages in the mouse embryo with key Transcription factors - Model relative to roles played by *Oct4*, *Nanog*, *Sox2*, *foxD3*, *Cdx2*, *Gata6* during early mouse preimplantation development. Pluripotent stem cells (green) are imaged in a morula as the inner cells, which then form the inner cell mass (ICM) of the blastocyst. Oct3/4 is essential in the first embryonic lineage specification. *Nanog* function is to prevent endoderm differentiation of ICM. *Sox2* and *FoxD3* are essential in the maintainance of a pluripotent epiblast. The figure is adapted from Niwa(2007) (4).

1.2 Properties of cellular decision making processes

All of the processes mentioned in the last section take time to completion. In early murine embryonic development for example, it takes about 1 week for stem cells to move from a pluripotent state to a differentiated state. Early stage differentiation actually starts after about 2 days (18). More so, the entire cellular decision process proceed in multiple steps controlled by different signals. For example the different stages on the way from a single stem cell to a specialized cell are controlled by different signals as shown in (Figure 1.3) modified from Alberts et al.(2008)(1). Cellular decision processes are controlled by complex signaling networks. For example the Wnt signaling pathway which plays a key role in the development of tissues and organs in multicellular organism involves a complex signaling mechanism taking place at the cell membrane, cytoplasm and inside the nucleus(Figure 1.4). Once the pathway is active traces in gene expression profiles can be observed reflecting the hierarchy of events along the pathway. My goal is to model the temporal interplay of signaling and expression in such complex biological processes involving several signaling pathways and spanning multiple rounds of cell signaling, gene regulation, and gene expression.

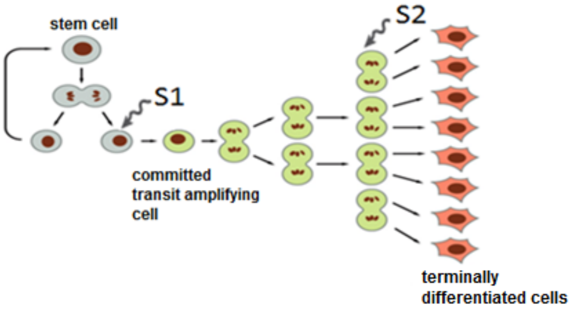


Figure 1.3: Properties of cell decision making - Cell decision making takes time and operate in multiple steps. Stem cell differentiation to a specialized cell fate involves a series of decision-making steps. Each decision making step is triggered by external or internal signals. The figure is a modification from (1).

1.3 Cellular decision making processes can be represented by hierarchies

Cellular decision processes can be modelled as hierarchies. A given hierarchy of signaling steps can be represented by a graph or network of upstream / downstream relations where nodes can be steps or controlling signaling genes and edges indicate upstream / downstream relations. Based on such a relationship we would expect a transitive closed graph. If S1 is upstream of S2 and S2 is upstream of S3, then by definition S1 is upstream of S3. Figure 1.5 shows a decision process comprising of 5 steps S1-S5. Steps S2-S5 can only occur after step S1 has occurred. For example, the MAP kinase cascade is activated by *Ras* which further leads to the activation of other important regulatory proteins such as *Myc*. So *Ras* acts upstream of *Myc* and this property is represented by a directed edge from *Ras* to *Myc*. The biological meaning of a network component depends on what kind of data we analyze. We mostly speak of network components as genes. However statistical methods available for gene regulatory networks can also be generalized for protein data (24, 25, 26).

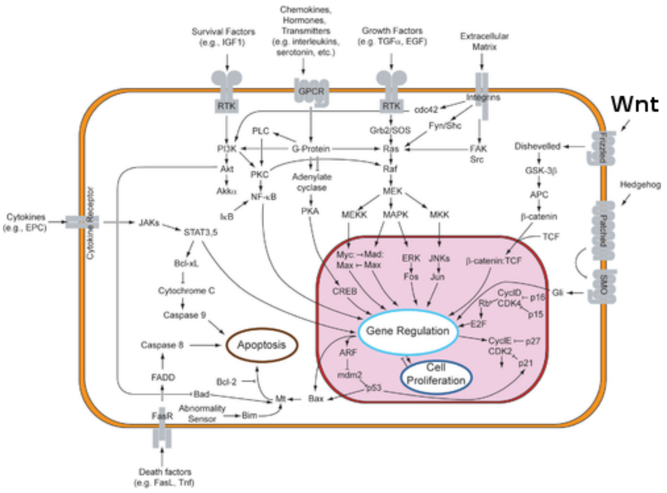


Figure 1.4: Properties of cell decision making - Wnt signaling involves a very complex signaling mechanism at the cell membrane, cytoplasm and nucleus. *Wnt* proteins bind to receptors on the cell surface to induce several intracellular signal transduction pathways. Through several cytoplasmic relay molecules, the signal is transduced to β-catenin, which enters the nucleus and forms a complex with *TCF* protein to activate transcription of Wnt target genes. This Figure has been taken from Wikipedia.org.

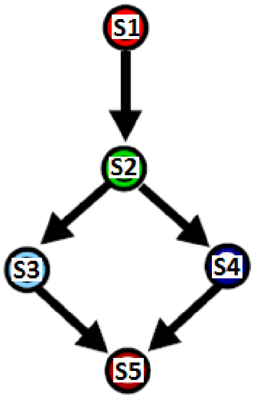


Figure 1.5: cellular decision processes can be modeled as hierarchies - The figure shows a cellular decision process comprising of 5 steps S1-S5. Steps S2-S5 can only occur after step S1 has occurred.

1.4 Statistical methods for analyzing decision making processes

With the advent of high throughput genomic technologies such as microarrays that can capture the expression of thousands of genes and the availability of powerful computational approaches, the practice of studying cellular processes at the systems level has greatly improved in the last decades. Numerous statistical methods have been suggested for the analysis and reconstruction of regulatory networks. Among the most widely used are relevance networks (27), graphical Gaussian models (28, 29), methods from information theory (30), Bayesian networks (31) including dynamic Bayesian networks (32, 33), Boolean networks (34, 35, 36) and methods based on ordinary differential equations (37, 38, 39). All these methods employ pure observational data, where the network was not perturbed experimentally. Relevance networks, graphical Gaussian models, Information theory approaches, Boolean networks, Bayesian and dynamic Bayesian networks are probabilistic in design motivated by the fact that signal transduction, gene expression and its regulation are stochastic processes (40, 41, 41). They mainly account for transcriptional effects in the cell. Apart from dynamic Bayesian and Boolean networks, they infer static transcriptional regulatory networks. Ordinary differential equations(ODEs) on the other hand are deterministic making strong assumptions on the network structure and interactions. The famous Michealis-Menten equation in the context of enzyme kinetics is an example (42). Similar to dynamic Bayesian networks ODEs allow for changes over time. A comprehensive overview of these methods in relation to transcriptional regulatory networks can be found in (7, 43).

1.4.1 Learning from experimental interventions

Simulation (44, 45) and experimental studies (24, 44) show that perturbation experiments greatly improve performance in network reconstruction. Rung *et al.* (46) built a directed disruption graph by connecting two genes where perturbation of the first gene resulted in expression changes in the other gene. However, disruption networks do not separate direct from indirect effects. Wagner (47) uses transitive reductions to find parsimonious subgraphs explaining a disruption network. The framework of Bayesian networks was also extended to account for perturbation data (48, 49). Yeang *et al.* search for topologies that are consistent with observed downstream effects of interventions (50). Although this algorithm is not confined to the transcriptional level of regulation, it requires that most signaling genes show effects when perturbing others.

The methods described exhaustively in this thesis build on Nested Effects Models (NEMs), which were first proposed by Markowetz *et al.* (49) for the analysis of non-transcriptional signaling networks. They differ from other statistical approaches like Bayesian networks or Boolean Networks by encoding subset relations instead of partial correlations. NEMs infer the graph of upstream/downstream relations for a set of signaling genes from perturbation effects. Since non-transcriptional signaling is too fast to be analyzed by delays of downstream effects, time series are not used. This changes when analyzing slow-going biological processes like cell differentiation.

1.5 Motivation for dynamic modeling of cell decision making processes

Note that there is a difference between the upstream/downstream relations of a network and the actual signal flow: If gene S_1 is upstream of S_2 and gene S_2 is upstream of S_3, consistency requires that S_1 is also upstream of S_3. While the consistency argument is valid for upstream/downstream relations, it does not hold for signal flows. Assume we have a linear cascade of signaling genes where the signal flows from S_1 *via* S_2 to S_3 (See Figure 1.6). Whether there is an alternative signal flow from S_1 directly to S_3 does not follow from upstream/downstream relations. However, evidence of the alternative signal flow comes from time delays of downstream effects. Assume that the time spent to propagate a downstream effect from S_1 to S_2 plus the time spent to propagate it from S_2 to S_3 is larger than the time to propagate the effect from S_1 to S_3 directly, then there must exist an alternative short cut pathway from S_1 to S_3. Thus, temporal expression measurements yield additional insight into the cellular signal flow.

1.5.1 The Feed-Forward Loop Network motif

Signaling networks that regulate the responses of living cells were recently found to obey recurring circuit modules that carry out key functions (3, 7, 51). They contain several biochemical wiring patterns, termed network motifs, which recur throughout the network (52). One of these motifs is the Feed-Forward Loop (FFL) (7). The FFL, a three-gene pattern, is composed of two input transcription factors(regulators), one

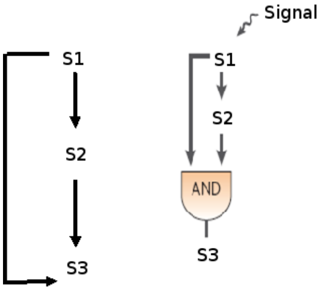


Figure 1.6: Transitive edge and Feed-Forward Loop - Transitive edges represent feed forward loops. The mode of interaction in the FFL can be activation or repression with an AND or OR input function at S_3. S_1 is activated by input signal(s).

of which regulates the other, both jointly regulating a target gene. The FFL has eight possible structural types (Figure 1.7) , because each of the three interactions in the FFL can be activating or repressing. Uri Alon and colleagues (3) found out that four of the FFL types, termed incoherent FFLs(Figure 1.7) act as sign-sensitive accelerators: they speed up the response time of the target gene expression following stimulus steps in one direction (e.g., off to on) but not in the other direction (on to off). The other four types, coherent FFLs, act as sign-sensitive delays. Thus they carry out specific dynamical functions. In addition each of the coherent and incoherent types of FFLs can have an AND or OR input function at the promoter of the target gene depending upon whether both or only one of the two regulators are needed to regulate the target gene (Figure 1.6). The transitive triple representation in Figure 1.6 shows that the transitive edge(S1S3) combine with the indirect edges(S1S2 and S2S3) to form a coherent feed forward loop(FFL). Nature uses these FFLs in several organisms to cause time delays so that the cell can function properly by filtering out random fluctuations. We may be able to reconstruct FFLs in a network if we can measure or estimate time delays in the signaling network.

1.5.2 Feed-Back Loop Network motif

Biological networks are all known to contain Feed-Back Loops(FBLs) and cycles (1, 7). For example in regulatory networks, TFs are known to be both negatively and positively auto-regulated. Negative auto-regulation occurs when a TF represses its own transcription. Such a simple circuit has been used to show the speed of response time

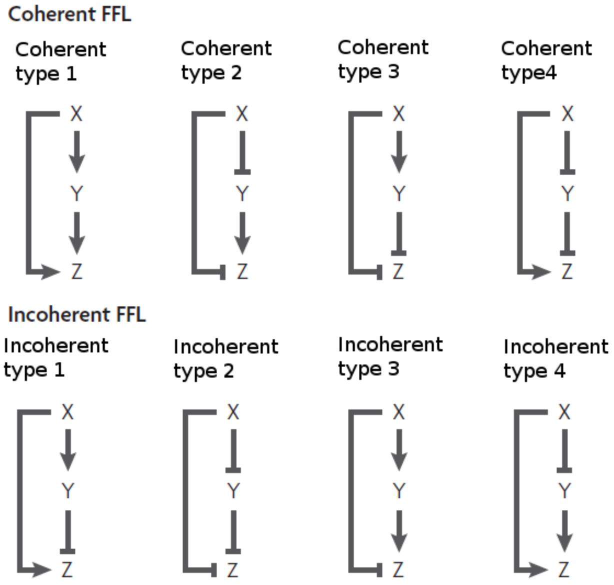


Figure 1.7: Feed-Forward Loops(FFLs) - The eight types of feedforward loops (FFLs) are shown. Arrows denote activation and ⊣ symbols denote repression. In coherent FFLs, the sign of the direct path from input factor X to output Z is the same as the overall sign of the indirect path through factor Y. Incoherent FFLs have opposite signs for the two paths. This figure is reproduced from (3).

and reduction of the cell-cell variation in protein levels (53, 54, 55). Similarly, positive auto-regulation occurs when a TF activates its own transcription by up-regulating itself. Positive auto-regulatory circuits have been shown to have opposite effects as to those of negative auto-regulatory feedback loops.(56). Modeling the cell cycle or autoregulation with an acyclic model (31) may not be the best idea due to loss of useful information. Fortunately, the cycle problem can be solved by assuming that the system evolves over time.

1.6 Thesis Organization

There are two main goals involve in analyzing time series RNA interference (RNAi) data for reverse engineering purposes. First, how to infer signaling pathways if direct observations of gene silencing effects on other network components may not be visible in the data. Second, to infer the signaling dynamics between pathway components from the data. This thesis summarizes methodology to address the first question and proposes a novel methodology to answer the second question. It is organized mainly as follows.

1.6.1 Nested Effects Models

Chapter 2 gives an overview of Nested effects models and their implementations. The theory of NEMs has been applied and extended in several studies. I give an in depth overview of all NEMs from literature in this chapter. Chapter 2 also works out the similarities, differences and limitations of all the methods.

1.6.2 Gibbs sampling and Nested Effects Models

In chapter 3, the concept of Gibbs sampling is reviewed. I discuss how such an estimation algorithm is used in several bioinformatics applications with particular enfancy on how it fits in within the context of Nested Effects Models.

1.6.3 Dynamic Nested Effects Models

In chapter 4, I develop a novel theory of learning from time series gene perturbations in the framework of Nested Effects Models (NEMs), called Dynamic Nested Effects Models(DNEMs). Chapter 4 goes further to demonstrate how DNEMs can be used to estimate time delays in a given network as well as make inferences on signal flow in a given network. The practical use is exemplified in an application to molecular mechanism in early stem cell differentiation. Finally a section on the speed up by stochasticity effects in dynamic networks is introduced as a by product of DNEMs.

1.6.4 Cyclic Dynamic Nested Effects Models

An extension of DNEM to handle cycles is discussed in chapter 5 with the help of simulations and an application to stem cell differentiation. In chapter 6 I discuss the impact of DNEMs by making a comparison to another complementary and faster modeling approach which also has the ability to unravel the regulatory networks across time.

2

Nested Effects Models

In modern biology, the key to inferring gene function and regulatory pathways are experiments with interventions into the normal functioning in a cell. A common technique is to perturb a gene of interest experimentally and to study which other genes activities are affected using gene expression monitoring. However, one of the key problems of analyzing perturbation screens is that the observed phenotypes occur downstream of the perturbed pathway and may not be able to show the direct influence of one particular pathway component on another. This is illustrated here by the cartoon pathway adapted from Wagner 2001(Figure 2.1) showing a cascade of five genes/proteins (S1-S5). Proteins S1-S3 form a kinase cascade, S4 is a transcription factor acting on S5. Up-regulation of S1 starts information flow in the cascade and results in S5 being turned on. In gene expression data this is visible as a correlation between S1 and S5(represented as an undirected edge in the model). Experimentally perturbing a gene, say S3, removes the corresponding protein from the cascade, breaks the information flow, and results in an expression change at S5 (represented as an arrow in the model). However, the different phosphorylation and activation states of proteins S2-S4 are not visible as changes in gene expression. Thus, because of the pathway mostly acting on the protein level most parts of the cascade (dashed arrows in the model) can not be inferred from gene expression data directly. One class of models that was developed to handle indirect information and high-dimensional phenotypes are ***Nested Effects Models*** (49).

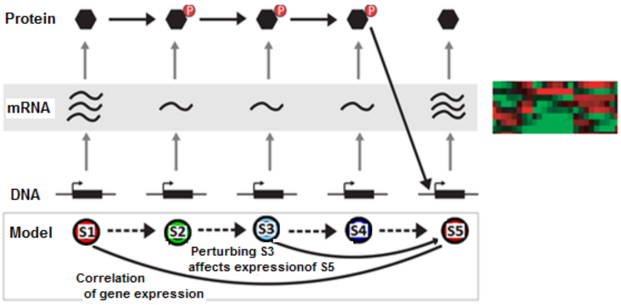


Figure 2.1: Cellular networks underlying observable phenotypes - Global molecular phenotypes like gene expression allow a view inside the cell but also have limitations. In gene expression data a correlation between proteins S1 and S5(represented as an undirected edge in the model) is visible. In addition, experimentally perturbing a gene, say S3, breaks the information flow, and results in an expression change at S5 (represented as an arrow in the model). However, the different phosphorylation and activation states of proteins S2-S4 are not visible as changes in gene expression. If the pathway is mostly acting on the protein level most parts of the cascade (dashed arrows in the model) can not be inferred from gene expression data directly. The figure is adapted from Wagner 2001.

2.1 Nested Effects Models(NEMs)

Following Markowetz *et al.* (49), we call the perturbed genes *S-genes* for signaling genes and denote them by $\mathbf{S} = S_1, \ldots, S_n$. The genes that change expression after perturbation are called *E-genes* and we denote them by $\mathbf{E} = E_1, \ldots, E_N$. We further denote the set of E-genes displaying expression changes in response to the perturbation of S_i by $\mathcal{D}_i$. In a nutshell: NEMs infer that S_1 acts upstream of S_2:

$$S_1 \longrightarrow S_2 \ \text{ if } \ \mathcal{D}_2 \subset \mathcal{D}_1$$

All downstream effects of a perturbation in S_2 can also be triggered by perturbing S_1 (Figure 2.2). This suggests that the perturbation of S_1 causes a perturbation of S_2 and acts upstream of S_2. In a general setting with more than two S-genes, we call the subset of S-genes, which are in an active state when S-gene S_j is silenced, the influence region of S_j. The set of all influence regions is called a silencing scheme Φ. It summarizes the effects of interventions predicted from the pathway hypothesis. This is mathematically represented as a transitively closed graph. The graph of upstream/downstream relations is estimated from the nested structure of downstream effects. Due to noise in the data, we do not expect strict super-/subset relations. Instead, NEMs recover rough nesting. Following Markowetz *et al.* (49) we assume only directed acyclic graphs. In the context of NEMs the most often used scoring metric is the posterior probability of a network Φ given data D, $P(\Phi|D)$. According to Bayes rule, the posterior probability can be written as

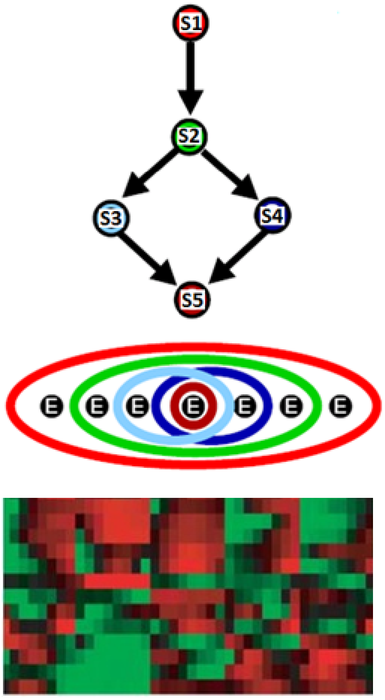


Figure 2.2: Cellular pathways can be reconstructed from the nested structure of downstream effects - If the target genes of S2 are a subset of the target genes of S1 then S1 acts upstream of S2. So in this sense all the target genes of S2 can be triggered by perturbing S1. Information on the target gene expressions can be obtained on a microarray.

$$P(\Phi|D) = \frac{P(D|\Phi)P(\Phi)}{P(D)}, \tag{2.1}$$

where $P(D)$ is a constant that does not depend on Φ. Consequently, the (marginal) likelihood $P(D|\Phi)$ together with the network prior $P(\Phi)$ play the central role in the inference.

In practice, we do not know which target genes or E-genes are being controlled by which S-genes. We first need to estimate the E-gene positions before scoring the graph. Closely following the presentation of Markowetz *et al.*(2005) (49) we denote the parameter for the E-gene positions as Θ. If we let $\Theta = \{\theta_i\}_{i=1}^{m}$ with $\theta_i \in \{1, ..., n\}$and $\theta_i = j$ if E_i is attached to S_j. In a perturbation experiment we predict effects at all E-genes, which are attached to an S-gene in the influence region. Expected effects can be compared with observed effects in the data to choose the topology, which fits the data best. Owing to measurement noise we cannot expect to find an expected topology to be in complete agreement with all observations. We allow deviation from predicted effects by introducing error probabilities α and β for false positive and negative situations, respectively. We model the expression levels of E-genes on the various perturbation experiments k as binary random variables E_{ik} . The distribution of E_{ik} is determined by the silencing scheme Φ and the error probabilities α and β. For all E-genes and targets of intervention, the conditional probability of E-gene state e_{ik} given silencing scheme Φ can then be written in tabular form as:

Table 2.1: The distribution of binary effect data - The distribution of E_{ik} is determined by the silencing scheme Φ and the error probabilities α and β.

$$\begin{array}{rcccl} & e_{ik}=1 & e_i=0 & & \\ \cline{2-3} P(e_{ik}|\Phi,\theta_i=j)=\Bigg\{ & \alpha & 1-\alpha & \text{if} \quad \Phi & \text{predicts } \textbf{no effect} \\ & 1-\beta & \beta & \text{if} \quad \Phi & \text{predicts } \textbf{effect} \end{array}$$

This means that if E_i is not in the influence region of the S-gene silenced in experiment k, the probability of observing E_{ik}=1 is α(probability of false alarm); the probability to miss an effect and observe $E_{ik} = 0$ even though E_i lies in the influence region is β (probability of missed signal). In the following, we summarize NEMs based on their statistical approach for dealing with Θ in scoring a given network.

2.1.1 Marginal likelihood scoring

In the Bayesian framework of Markowetz *et al.*(2005) (49), networks are scored by marginal posterior probabilities which depend on the marginal likelihood of the parameter space. The marginal likelihood involves marginalization over the whole parameter space Θ .

$$P(D|\Phi) = \int_{\Theta} P(D|\Phi, \Theta)P(\Theta|\Phi)d\Theta. \tag{2.2}$$

The marginal likelihood $P(D|\Phi, \Theta)$ is based on the following assumptions given in (49):

1. Given silencing scheme Φ, and fixed positions of E-genes Θ, the observations in D are sampled independently and distributed identically:

   $$P(D|\Phi,\Theta) = \prod_{i=1}^{m} P(D_i|\Phi,\theta_i) = \prod_{i=1}^{m}\prod_{k=1}^{l} p(e_{ik}|\Phi,\theta_i), \tag{2.3}$$

   where D_i is the ith row in data matrix D.

2. Parameter independence. The position of one E-gene is independent of the positions of all the other E-genes at any given time:

   $$P(\Theta|\Phi) = \prod_{i=1}^{m} P(\theta_i|\Phi) \tag{2.4}$$

3. Uniform prior. The prior probability to attach an E-gene is uniform over all S-genes:

   $$P(\theta_i = j|\Phi) = \frac{1}{n} \qquad \text{for all} \quad i \quad \text{and} \quad j \tag{2.5}$$

   However prior existing biological knowledge about regulatory modules can be incorporated (57, 58).

With the above assumptions the marginal likelihood can be calculated thus. The numbers above the equality sign indicate which assumption was used in each step.

$$\begin{aligned} P(D|\Phi) &= \int_{\Theta} P(D|\Phi,\Theta)P(\Theta|\Phi)d\Theta \\ &\overset{[1,2]}{=} \prod_{i=1}^{m}\int_{\theta_i} P(D_i|\Phi,\theta_i)P(\theta_i|\Phi)d\theta_i \\ &\overset{[3]}{=} \frac{1}{n^m}\prod_{i=1}^{m}\sum_{j=1}^{n} P(D_i|\Phi,\theta_i = j) \\ &\overset{[1]}{=} \frac{1}{n^m}\prod_{i=1}^{m}\sum_{j=1}^{n}\prod_{k=1}^{l} p_{\alpha,\beta}(e_{ik}|\Phi,\theta_i = j) \end{aligned} \tag{2.6}$$

Note here that we can sum over all E-gene positions since we have a finite number of S-genes.

2.1.2 Maximum likelihood scoring

It is also possible to maximize the joint posterior distribution of Φ and Θ. We can represent both the silencing scheme and the E-gene positions as adjacency matrices whose entries represent edges between S-S-genes and S-E-genes respectively. For the purpose of consistency we denote both matrices as Φ and Θ. Tresch (59) defined a Nested Effects Model (NEM) F as a product of Φ and Θ:

$$F = \Phi\Theta \tag{2.7}$$

Using the same formulation as in the previous section and assuming data independence, the likelihood of the model F is represented as $P(D|F)$ and factors out as follows:

$$\begin{aligned} P(D|F) &= P(D|\Phi\Theta) \\ &= \prod_{(j,i)\in\Phi\times\Theta} P(D_{j,i}|j = F_{ji}) \\ \mathit{or}\log(P(D|F)) &= \sum_{(j,i)\in\Phi\times\Theta} logP(D_{j,i}|j = F_{ji}) + const, \end{aligned} \tag{2.8}$$

if we assume equal probability for observing a 1 or 0 and $(j, i) \in \Phi \times \Theta$ with $j = F_{ji}$ interpreted as S-gene j is linked to E-gene i . The quantity $\log(P(D|F))$ can also be expressed as a likelihood ratio for convenience using matrix algebra as follows:

$$\log(P(D|F)) - \log(P(D|N)) = tr(FR), \tag{2.9}$$

where $R = \log \frac{P(D_{ji}|e_{ij}=1)}{P(D_{ji}|e_{ij}=0)}$, "$tr$" denoting the trace function of a quadratic matrix and N the NULL matrix corresponding to the model predicting no effects at all. Hence the marginal likelihood of the data becomes

$$\log(P(D|\Phi, \Theta)) = tr(\Phi\Theta R) + const \tag{2.10}$$

This form provides a flexible way of handling input data binary values, p-values, or any other arbitrary statistic as long as it can be converted to a likelihood ratio. The aim of the NEMs is to find the optimal silencing scheme $\hat{\Phi}$. The posterior model (Φ, Θ) written in log form is

$$\log(P(\Phi, \Theta|D)) = \log(P(D|\Phi, \Theta)) + \log(P(\Phi)) + \log(P(\Theta)) + const, \tag{2.11}$$

and the task is to find the maximum aposteriori(MAP) estimate for $\log(P(\Phi,\Theta|D))$,

$$(\hat{\Phi},\hat{\Theta}) = \text{argmax}_{\phi,\Theta}(\log(P(D|\Phi,\Theta)) + \log(P(\Phi)) + \log(P(\Theta))), \tag{2.12}$$

So in order to maximize the graph we need to maximize the E-gene positions and *vice versa.*

2.1.3 NEMs as a Bayesian network

Zeller *et al.* (60) introduced a Bayesian network view on NEMs. A Bayesian network is defined by a graphical model structure Φ and a family of conditional distributions $F^{'}$ and their parameters $\boldsymbol{\Theta}$ (61, 62). The model structure Φ consists of a set of nodes V and a set of directed edges E connecting the nodes such that the resulting directed graph is acyclic (DAG). The nodes represent random variables in the network whereas the edges encode a set of conditional dependencies. In the parametric setting, the family of conditional distributions $F^{'}$ is assumed to be known and hence is fully described by its parameters. Let $\mathbf{X} = X_1, X_2, ..., X_n$ denote a set of random variables that correspond to the nodes V in the network. Lower-case letters $x_1, x_2, ..., x_n$ are used to denote the value of the corresponding variables. Let $\mathbf{Pa}(X_i)$ denote the random variables corresponding to the parents of node i in the DAG. Then, the network structure Φ and the parameters $\boldsymbol{\Theta}$ of the conditional distributions together define a joint distribution over the random variables $\mathbf{X}$ as

$$P(x_i, x_2, ..., x_n) = \prod_{i=1}^{n} P(x_i|\mathbf{pa}(X_i)) \tag{2.13}$$

In the context of NEMs following the presentation in (59), we have to model a deterministic signaling hierarchy, in which some components (E) can be probed by measurements, and some components (S) are perturbed in order to measure the reaction of the system as a whole. Let $H^{'}$ be the nodes of an acyclic graph representing a combination of the S-S-genes and S-E-genes connection Figure 2.3. A,B,C represent the S-genes and $X_1, X_2, Y_1, Y_2, Z_1, Z_2$ represent the effect nodes. We assume $H^{'}$ as hidden in the sense that no observation will be available for $H^{'}$. In order to account for the data, we introduce an additional layer of observable variables $(observables, O)$ in the following way: each effect node $e \in E$ has an edge pointing to a unique observable node $e^{'} \in O$ (Figure 2.3). Hence, $O = \{e^{'}|e \in E\}$, and we call $e^{'}$ the observation of e. Similar like

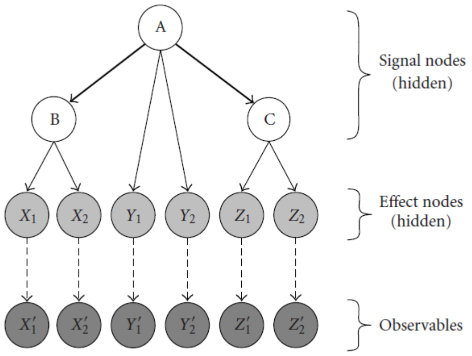


Figure 2.3: Bayesian Nested effects models - Example of a Nested effects model in its Bayesian network formulation. The bold arrows determine the graph Φ, the solid thin arrows encode Θ. Dashed arrows connect the effects to their reporters. This figure is reproduced from (59).

before we let $pa(x)$ be the set of parents of a node x and for notational convenience add a zero node z, $p(z = 0) = 1$, which has no parents, and which is a parent of all hidden nodes (but not of the observable measurements). For the hidden nodes, define local probabilities corresponding to deterministic relationships as follows :

$$\begin{aligned} p(x = 1|pa(x)) &= \begin{cases} 1 & \text{if any parent is active} \\ 0 & \text{otherwise,} \end{cases} \\ &= \max(pa(x)) \text{ for } x \in H^{'}, \end{aligned} \tag{2.14}$$

All hidden nodes are set to 0 or 1 deterministically, given their parents. The local probabilities $p(e^{'}|e \in E)$, $e \in E$ can come from both discrete or continuous distributions (59). From Equation 2.13 the Bayesian network NEM is parameterized by its topology Φ and its local probability distributions, which we assume to be given by a set of local parameters Θ. The ultimate goal is to maximize $P(\Phi|D)$. In the presence of prior knowledge and if we assume independent priors for the topology and the local parameters, we can write

$$\begin{aligned} P(\Phi,\Theta|D) &= \frac{P(D|\Phi,\Theta)P(\Phi)P(\Theta)}{P(D)} \\ &\propto P(D|\Phi,\Theta)P(\Phi)P(\Theta) \end{aligned} \tag{2.15}$$

from which it follows that

$$\begin{aligned} P(\Phi|D) &= \int P(\Phi,\Theta|D)d\Phi \\ &\propto P(\Phi)\int P(D|\Phi,\Theta)P(\Theta)d\Theta \end{aligned} \tag{2.16}$$

Its difficult to solve the integral analytically. We resort to a simultaneous maximum a posteriori estimation of Φ and Θ (59). Thus

$$\begin{aligned} (\hat{\Phi},\hat{\Theta}) &= \text{argmax}_{\Phi,\Theta}P(\Phi,\Theta|D) \\ &= \text{argmax}_{\Phi}(\text{argmax}_{\Theta}P(D|\Phi,\theta)P(\Theta))P(\Phi). \end{aligned} \tag{2.17}$$

2.1.4 Factor graph NEMs

Finally, a signed version of the Nested Effects Model and an associated efficient structure inference method, named Factor Graph-Nested Effects Model(FG-NEM) (2) was developed to distinguish between activating and inhibiting regulation in a pathway. Recall that the original NEM by Markowetz *et al.* (49) include two sets of parameters. The parameter set Φ records all pair-wise interactions among the S-genes and the parameter set Θ describes how each E-gene is attached to the network of S-genes. Φ is a binary matrix with entry ϕ_{AB} set to one if S-gene A acts above S-gene B and zero otherwise. Φ must also be transitively closed. The model by Markowetz *et al.* (49) does not distinguish between stimulatory and inhibitory interactions. To tackle this drawback, Vaske *et al.* (2) suggest a model, in which ϕ_{AB} takes six possible values for each unique unordered S-gene pair A,B also known as interaction modes. The possible values are: 1) A activates B, $A \rightarrow B$; 2) A inhibits B,$A \dashv B$; 3) A is equivalent to B, A=B; 4) A does not interact with B, $A \neq B$; 5) B activates A,$B \rightarrow A$; and 6) B inhibits A, $B \dashv A$. The Factor graph NEMs allow for the reconstruction of a broader set of S-gene interactions from the secondary effects of E-gene expression corresponding to the observed data denoted as D. Similarly like the other NEM approaches discussed so far, a maximum aposteriori is used to identify the Φ that maximizes the posterior

$P(\Phi|D)$ represented as :

$$\begin{aligned}\hat{\Phi} &= \text{argmax}_{\Phi} P(\Phi|D) \\ &= \text{argmax}_{\Phi} \sum_{\Theta}\sum_{H} P(\Phi, \Theta, H|D). \end{aligned} \tag{2.18}$$

where Θ refers to the attachment point of each E-gene into the network and H refers to the hidden E-gene states corresponding to up, down regulations or no change. Applying the same assumptions as in Markowetz *et al.* (49) we have:

$$\begin{aligned}\hat{\Phi} &= \text{argmax}_{\Phi} P(\Phi) \sum_{\Theta} P(\Theta|\Phi) \sum_{H} P(H|\Phi, \Theta) P(D|H) \\ &= \text{argmax}_{\Phi} P(\Phi) \sum_{\Theta}\sum_{H} P(H|\Phi, \Theta) P(D|H) \\ &= \text{argmax}_{\Phi} P(\Phi) \prod_{e \in E} \sum_{\Theta}\sum_{H} P(H_e|\Phi, \theta_e) P(D_e|H_e) \end{aligned}$$

given independence of E-genes, E.

$$= \text{argmax}_{\Phi} P(\Phi) \prod_{e \in E} L_e(\Phi) \tag{2.19}$$

where D_e and H_e are the row vectors of data matrix and hidden states for E-genes respectively and θ_e records the attachment of an E-gene to an S-gene and L_e summarizes the marginal likelihood of the data restricted only to E-gene e under a given model Φ and θ_e. Note that L_e can be reformulated as a product of pair-wise S-gene terms(2).

2.1.4.1 Structure of factor graph NEMs and Network inference

Scoring a given S-gene graph can be achieved based on max-sum message passing in a factor graph (63) which provides an efficient means for estimating highly probable S-gene configurations. The parameters that determine the S-gene interactions, Φ, are explicitly represented as variables in the factor graph. Identifying a high-scoring S-gene network is therefore converted to the task of identifying likely assignments of the Φ variables in the factor graph. A factor graph is a probabilistic graphical model whose likelihood function can be factorized into smaller terms (factors) representing local constraints on a set of random variables. A factor graph can be represented as an undirected, bi-partite graph with two types of nodes: variables and factors. A variable is adjacent to a factor if the variable appears as an argument of the factor. Figure 2.4 shows the factor graph representation of a Bayesian network. Factor graphs represent both the variables as nodes and the factors as nodes, with edges from each factor to the variables in that factor's domain, resulting in a bipartite graph. Factor graphs generalize probability mass functions as the joint likelihood function requires no normalization and the factors need not be conditional probabilities. Each factor encodes a local constraint pertaining to a few variables. In the factor graph NEM a Φ that maximizes the posterior is found using max-sum message passing using all terms from Equation 2.19 in log space. The complete model of a factor graph NEM by Vaske *et al.*(2009) contains three types of variables and three classes of factors. The variables include: the continuous random observation of E-gene expression under a given intervention and replicate experiment, the unknown hidden state of E-gene under a particular intervention which is a discrete variable with domain $\{1, 0, -1\}$ and the interaction modes between two S-genes. The factors consists of: the Expression factors which model expression as a mixture of Gaussian distributions, the Interaction Factors which constrain E-gene states to five possible types of interaction modes between two S-genes and the Transitivity factors which constrain pair-wise interactions to form consistent triplets of S-genes. During message passing, messages which are simply local belief potentials associated with variable interactions are passed between all nodes(variables) in the

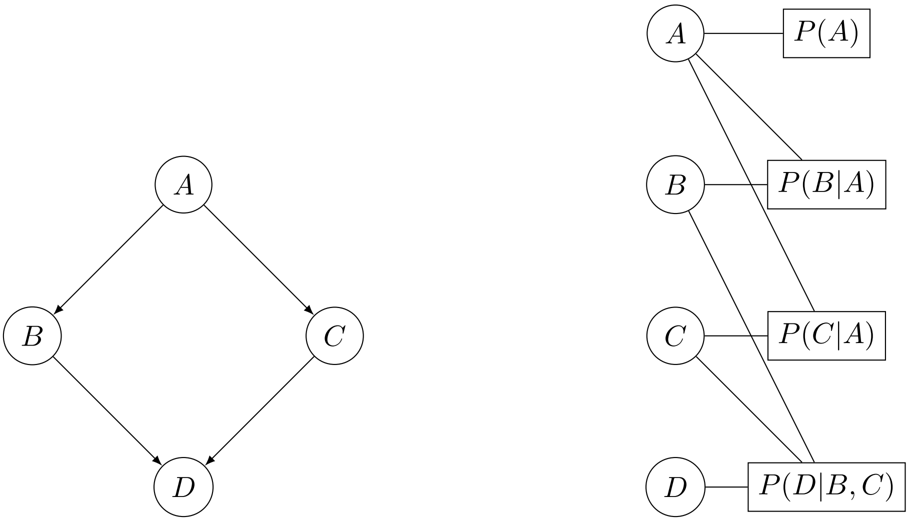


Figure 2.4: Bayesian network next to corresponding factor graph- A Bayesian network (left) and the corresponding factor graph (right). The decomposition of the joint probability, $\mathrm{P}(A,B,C,D) = \mathrm{P}(D|B,C)\ \mathrm{P}(B|A)\ \mathrm{P}(C|A)\ \mathrm{P}(A)$ is made explicit in the bipartite factor graph.

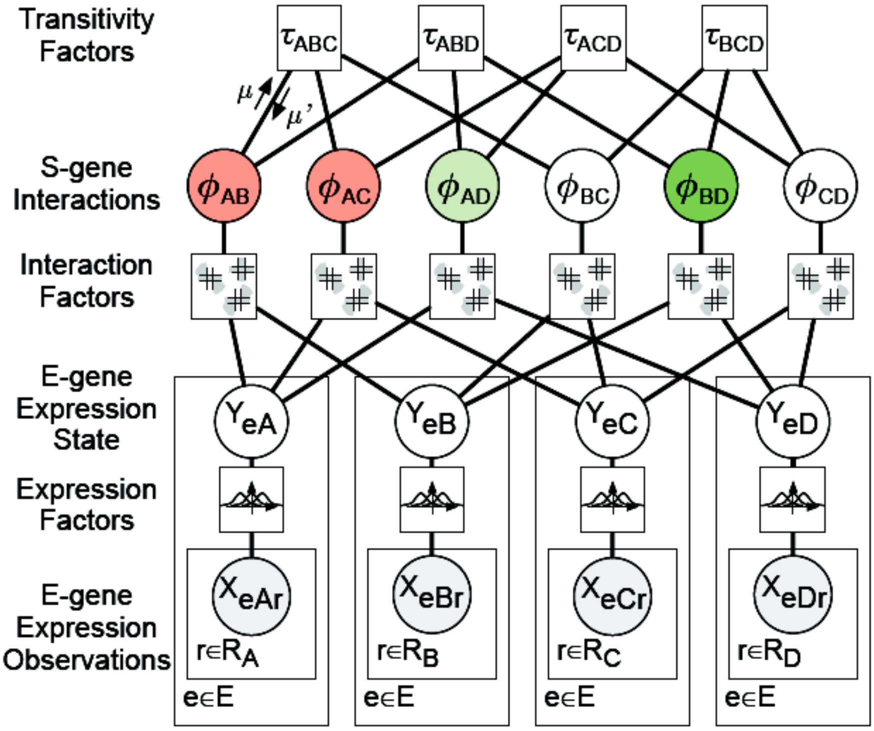


Figure 2.5: Structure of factor graph for network inference in Factor graph NEMs - The factor graph consists of three classes of variables (circles) and three classes of factors (squares). X_{eAr} is a continuous observation of E-gene e's expression under intervention A and replicate r. Y_{eA} is the hidden state of E-gene e under intervention A, and is a discrete variable with domain $\{1, 0, -1\}$. ϕ_{AB} is the interaction between two S-genes A and B. In this figure red, green and white shading denotes activation, inhibition and no interaction respectively. Expression Factors model expression as a mixture of Gaussian distributions. Interaction Factors constrain E-gene states to interaction modes between two S-genes. Transitivity Factors constrain pair-wise interactions to form consistent triangles. The arrows labeled μ and μ' are messages encoding local belief potentials on ϕ_{AB} and are propagated during factor graph inference. This figure is reproduced from (2).

graph using two inference steps. In the first step, messages from observation nodes are passed through the expression factors and hidden E-gene state variables, to calculate all messages in a single upward pass . In the second step, messages are passed between only the interaction variables and transitivity factors until convergence using Equation 2.19. The final S-gene network is derived by transitive reduction of all redundant edges from Φ. Figure 2.5 reproduced from (2) gives an overview of the structure of factor graph NEM with expression factors, interaction factors, and transitivity factors. For acyclic factor graphs, the marginal, max-marginal and conditional probabilities of single or multiple variables can be calculated exactly with the max-sum algorithms (63). Message-passing algorithms have been shown to demonstrate excellent empirical results in various practical problems even on graphs containing cycles such as feed-forward and feed-back loops(64, 65, 66).

2.2 Network learning algorithms in NEMs

Recall in the Bayesian framework of Markowetz *et al.*(2005) (49), networks are scored by posterior probabilities. By enumerating all network topologies, the maximum posterior network is selected. The exhaustive search limits the method to small networks of up to 6 S-genes. Thus, exhaustive enumeration is infeasible even for medium-sized studies. For large-scale screens, search heuristics are used to explore model space. Several approaches to this problem have been proposed by Frölich *et al.* (2, 57, 58, 59), all of which concentrate on small sub-models involving only pairs, triples, or quadruples of nodes. The final S-gene graph is scored by combining the scores from these sub models.

2.2.1 Pairwise and triple search

The division into subgraphs can either be into all pairs or triples of nodes (57). In the pairwise approach, for every pair of S-genes (A,B), we compute the Bayesian score detailed in section 2.1 and select the maximum aposteriori (MAP) model $M_{A,B} \in \{A \rightarrow B, B \rightarrow A, A = B, A \neq B\}$. The advantage of this approach is the increase in speed and the possibility to infer networks involving a very large number of nodes. However, the reconstruction accuracy of networks based on the pairwise method is rather low due to the pairwise independence assumption used in scoring the network which is not true in real biological networks. To improve on this limitation the triple search approach was introduced (57). This algorithm scores all 29 possible transitive edge interactions between 3 S-genes, selects the MAP model and combines these models into one final graph with the help of model averaging and thresholding. Even though all triplet models are transitively closed, edgewise model averaging and thresholding are not guaranteed to yield a transitively closed graph. An approximate transitive network among the S-genes can be computed by using the approach of Jacob *et al.*(2008) (67).

2.2.2 Module networks

Another divide-and-conquer approach by Frölich *et al.*(2007) (58) enable the analysis of larger networks with hundreds of S-genes. They divide the graph into smaller units called *modules*, use exhaustive enumeration for each subgraph, and then re-assemble the complete network. The division into subgraphs can either be into all pairs or triples of nodes (57) or a data-dependent approach into coherent modules (58) using alternative suitable clustering algorithms. The idea behind the latter is that S-genes with a similar E-gene response profile should be close in the signaling pathway. These modules are eventually further subdivided into smaller submodules until each submodule contains only 4 S-genes at most. The exhaustive search approach is then applied independently on these submodules and the optimal subnetworks are reconnected using pairwise node testing as well as transitive closure until the topology for the total network is completed.

2.2.3 Greedy hill search

Greedy search heuristics (58, 59) starts from an initial graph usually with no edges and then successively adds those edges, which increase the likelihood of the data most. Alternatively, starting with an initial estimate of the linking of E-genes to S-genes from the data, one can also perform an alternating MAP optimization of the S-genes graph and the linking graph until convergence. As a final step a transitively closed graph most similar to the original one can be estimated using transitive approximations(67).

2.3 Overview and differences on Nested effects models

Figure 2.6 organizes all the NEMs methods into a decision tree with respect to the following basic questions: Does the data include gene knockin or knockdown experiments or both? Does each experiment type involve single or multiple knockdown observations? If the former is true, does the model allow for changes over time? If yes, we call it dynamic, else static. Does the dynamic model include cycles? If yes we call dynamic cyclic NEMs, else dynamic non cyclic NEMs. Furthermore, does the dynamic cyclic and non-cyclic NEMs include a discrete or continuous model and finally are the scoring of these models based on a Marginal likelihood approach, Maximum likelihood method or full Bayesian methodology? In the leaf nodes of the decision tree methods that involve static models have been grouped together. Some branches in the tree are missing corresponding to areas where methodology has not yet been established so far although similar decisions can also be made. The three most left leaf nodes of Figure 2.6 highlights the main contribution of this dissertation. The dynamic NEMs extend NEMs to infer both the network structure as well as the dynamics of the network. It goes further to make inferences on feed back loops if they exist.

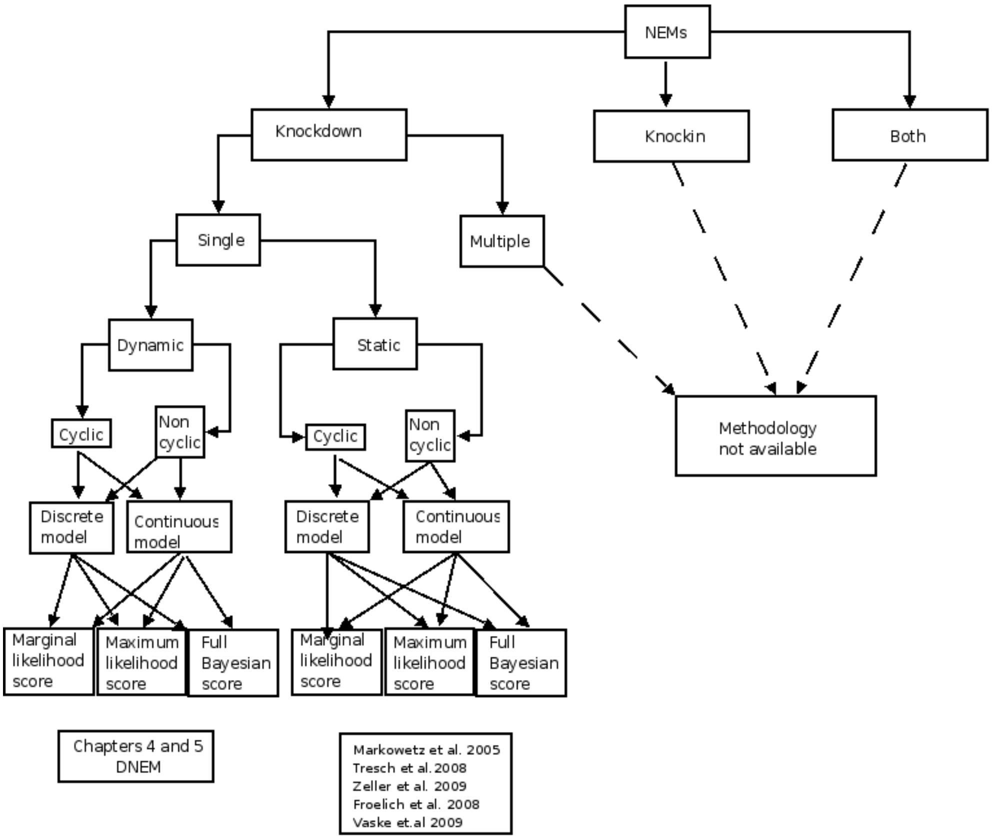


Figure 2.6: A guide to the literature on NEMs - The methods discussed in this chapter all fall into the right branch of node denoted as “single” corresponding to methodology for single knockdown data. The next two chapters will deal with learning the dynamics of a network, improve on accuracy of network reconstruction and making inferences on cyclic networks. The main contribution of this dissertation is modeling the temporal interplay of molecular signaling and gene expression using dynamic nested effects models from perturbation experiments.

3

Gibbs sampling

In this chapter I review the concept involved in Gibbs sampling and its use in bioinformatics. I go further to motivate its use for parameter estimation in learning the dynamics of a network.

3.1 Background on Gibbs sampling

Gibbs sampling also called *alternating conditional sampling* (68) is an algorithm to draw samples from a joint distribution based on the full conditional distributions of all the associated random variables. Though the idea originated from Hastings work in 1970 (69), it was first named Gibbs sampling by Geman and Geman (1984)(70) in a discussion of applications to image processing. Later on several statisticians became interested following the works of Tanner and Wong in 1987 (71) on use of iterative simulation in data augmentation and Gelfand and Smith in 1990 (72) on the uses of the Gibbs sampler in various Bayesian inference applications. Since then this algorithm has been used to estimate both posterior distributions as well as likelihood estimation in several domains. In particular it has been a popular alternative to Expectation-Maximization (EM) (73) for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the models depend on unobserved hidden variables. EM is a numerical maximization procedure that climbs in the likelihood space aiming to find the model parameters and the hidden variables that maximize the likelihood function. In contrast Gibbs sampling provides the means to estimate the target joint distribution of the hidden and known parameter space as a whole.

Maximum aposterior(MAP) estimates are often used. Gibbs sampling suffers less from the global and local maxima problems than the EM (74). This property makes it a suitable algorithm for solving model based problems that occur in bioinformatics, where the objective likelihood function is usually multimodal due to the high complexity of the data. Gibbs sampling is well established for the motif finding problem in DNA sequence analysis (75, 76, 77, 78). In this thesis we focus on its application to NEMs, more specifically its use in the estimation of parameters in the dynamic model of NEMs. In the following, we will first review the working mechanism of Gibbs sampling and then focus on its role in dynamic NEMs.

3.2 Gibbs sampling

Gibbs sampling avoids the cumbersome and sometimes non-trivial mathematical calculations of integrals in obtaining the joint distribution of a set of random variables, by sampling directly from their full conditional distributions. Since the same mechanism applies to both models for discrete data and models for continuous data, I use the terms "distribution" and "density" interchangeably. Suppose that we want to draw samples for the set of random variables $y_1, y_2, ..., y_K$, but that the marginal distributions and thus the joint distribution are too complex to directly sample from. Furthermore, assume that the full conditional distributions $p(y_i|y_j; j \neq i)$ (for $i = 1, ..., K$) are available. Starting from initial values $y_1^{(0)}, y_2^{(0)}, ..., y_K^{(0)}$, the Gibbs sampler draws samples of the random variables in the following order:

$$
\begin{array}{rcl}
y_1^{(t+1)} & \sim & p(y_1|y_2 = y_2^{(t)}, ..., y_K = y_K^{(t)}) \\
y_2^{(t+1)} & \sim & p(y_2|y_1 = y_1^{(t+1)}, y_3 = y_3^{(t)}, ..., y_K = y_K^{(t)}) \\
& \vdots & \\
y_i^{(t+1)} & \sim & p(y_i|y_1 = y_1^{(t+1)}, ..., y_{i-1} = y_{i-1}^{(t+1)}, y_{i+1} = y_{i+1}^{(t)}, ..., y_K = y_K^{(t)}) \\
& \vdots & \\
y_K^{(t+1)} & \sim & p(y_K|y_1 = y_1^{(t+1)}, ..., y_{K-1} = y_{K-1}^{(t+1)}),
\end{array}
\tag{3.1}
$$

where t denotes the iterations. Geman and Geman (1984)(70) shows that as $t \rightarrow \infty$, the distribution of $(y_1^{(t)}, ..., y_K^{(t)})$ converges to that of $(y_1, ..., y_K)$. Equivalently, as $t \rightarrow \infty$, the distribution of $y_i^{(t)}$ converges to $p(y_i)$ (for $i = 1, ..., K$).

3.2.1 The Markov chain property

The convergence of samples drawn by the Gibbs sampler relies on the fact that these samples form Markov chains. i.e. $((y_1^{(1)}, ..., y_K^{(1)}), .., (y_1^{(t)}, ..., y_K^{(t)}))$ as well as $(y_i^{(1)}, ..., y_i^{(t)})$ are Markov chains, where $(y_1^{(t)}, ..., y_K^{(t)})$ and $y_i^{(t)}$ are called the states of $y_1, y_2, ..., y_K$ and y_i respectively. The basic Markov chain property for a particular variable y_i for example is given as

$$P(y_i^{(t+1)}|y_i^{(t)}, ..., y_i^{(0)}) = P(y_i^{(t+1)}|y_i^{(t)}), \tag{3.2}$$

which means that the future state of a random variable depends only on its current state but not on on its past states. If

$$\begin{aligned} \pi_b(t+1) &= p(y_i^{(t+1)} = b) \\ \pi_a(t) &= p(y_i^{(t)} = a) \\ \text{and } p(a \rightarrow b) &= p(y_i^{(t+1)} = b|y_i^{(t)} = a), \\ \text{then} & \\ \pi_b(t+1) &= p(a \rightarrow b)\pi_a(t). \end{aligned} \tag{3.3}$$

$p(a \rightarrow b)$ is known as the transition probability of going from state a to b for random variables y_i. The probability transition matrix $\mathbf{P}$ is obtained by enumerating all the possible states for y_i along the rows and the columns, and then filling up the entire matrix with all the transition probabilities. Therefore, each row of $\mathbf{P}$ must sum to 1. Hence Equation 3.3 generalizes to

$$\pi(t+1) = \mathbf{P}(a \rightarrow b)\pi(t). \tag{3.4}$$

It has been shown that if all the entries of $\mathbf{P}$ are greater than 0, an evolving Markov chain will reach a stationary distribution π^* after a sufficient amount of time (79), i.e.,

$$\pi^* = \mathbf{P}\pi^*. \tag{3.5}$$

Casella and George(1992)(79) gives an intuitive proof that the stationary distributions of the Markov chains generated by Gibbs sampling are the joint distribution $p(y_1, y_2, ..., y_K)$ and the marginal distributions $p(y_i)$, and that the probability transition matrices of these Markov chains can be derived from the full conditional distributions.

3.2.2 The Monte Carlo property

To estimate the joint(or marginal) distribution only those samples collected after convergence can be used. The period which the samples are collected before convergence is reached is known as the "burn-in period " and the period after convergence during which samples are collected is known as "sampling period ". The samples collected in the sampling period enable us to calculate the expectation of a function $f(y_i)$ over the distribution $p(y_i)$. This is done by the Monte Carlo integration (80) given as :

$$E_{p(y_i)}[f(y_i)] = \int f(y_i) \cdot p(y_i) dy \approx \frac{1}{T} \sum_{t=1}^{T} f(y_i^{(t)}), \tag{3.6}$$

where t indexes the iterations in the sampling period, and T is the total number of samples collected. Thus, the expected value of y_i is calculated as

$$E_{p(y_i)}[y_i] = \int y_i \cdot p(y_i) dx \approx \frac{1}{T} \sum_{t=1}^{T} y_i^{(t)}. \tag{3.7}$$

Alternatively, a more accurate estimate of the expected value of y_i provided by Gelfand and Smith (1990) (72) using the Rao-Blackwell theorem (81)is given as:

$$E_{p(y_i)}[y_i] = \frac{1}{T} \sum_{t=1}^{T} E_{p(y_i|y_1^{(t)},...,y_{i-1}^{(t)},y_{i+1}^{(t)},...,y_K^{(t)})}[y_i]. \tag{3.8}$$

Similarly, the posterior distribution itself can be approximated by

$$E[p(y_i)] = \frac{1}{T} \sum_{t=1}^{T} p(y_i|y_1^{(t)}, ..., y_{i-1}^{(t)}, y_{i+1}^{(t)}, ..., y_K^{(t)}). \tag{3.9}$$

Hence, equation 3.6 can be generalized as:

$$E_{p(y_i)}[f(y_i)] = \frac{1}{T} \sum_{t=1}^{T} E_{p(y_i|y_1^{(t)},...,y_{i-1}^{(t)},y_{i+1}^{(t)},...,y_K^{(t)})}[f(y_i^{(t)}]. \tag{3.10}$$

The estimators obtained by Monte Carlo integration are unbiased estimators.

3.2.3 Variations of Gibbs sampling

Several different adaptations of Gibbs sampling exist. The primary purpose of these variations is to reduce autocorrelation (see subsection 3.2.6) between samples.

3.2.3.1 Blocked Gibbs sampler

This approach groups two or more variables together and samples from their joint distribution conditioned on all other variables, rather than sampling from each one individually. For example, in a hidden Markov model(82), a blocked Gibbs sampler might sample from all the hidden variables making up the Markov chain in one go, using the forward-backward algorithm (82), an inference algorithm which computes the posterior marginals of all hidden state variables given a sequence of observations.

3.2.3.2 Collapsed Gibbs sampler

This second alternative, integrates out (marginalizes over) one or more variables when sampling for some other variable. For example, imagine that a model consists of three variables X, Y, and Z. A full Gibbs sampler would sample from $p(X|Y,Z)$, then $p(Y|X,Z)$, then $p(Z|X,Y)$. A collapsed Gibbs sampler might replace the sampling step for X with a sample taken from the marginal distribution $p(X|Z)$, with variable Y integrated out in this case. Alternatively, variable Y could be collapsed out entirely, alternately sampling from $p(X|Z)$ and $p(Z|X)$ and not sampling over Y at all. The distribution over a variable X that arises when collapsing a parent variable Y is called a compound distribution; sampling from this distribution is generally tractable when Y is the conjugate prior for X, particularly when X and Y are members of the exponential family. For more information, see the article on compound distributions by Liu (83).

3.2.3.3 Gibbs sampler with ordered overrelaxation

In this variation, the Gibbs sampler samples a given odd number of candidate values for $y_i^{(t)}$ at any given step and sorts them, along with the single value for $y_i^{(t-1)}$. If $y_i^{(t-1)}$ is the s^{th} smallest in the sorted list then the $y_i^{(t)}$ is selected as the s^{th} largest in the sorted list. For more information, see Neal(1995)(84).

3.2.4 Extensions of Gibbs sampling

It is also possible to extend Gibbs sampling in various ways. For example, in the case of variables whose conditional distribution is not easy to sample from, a single iteration of slice sampling (85) or the Metropolis-Hastings algorithm (69, 86) can be used to sample from the variables in question. It is also possible to incorporate variables that are not random variables, but whose value is deterministically computed from other variables. Generalized linear models(87), e.g. logistic regression, can be incorporated in this fashion.

3.2.5 Assessing convergence

A very pertinent issue in using Gibbs sampling is to determine when the procedure has reached convergence. The number of iterations needed for the burn-in period varies from case to case. Depending on how dependent the early samples are and how strong the between and within sequence correlation is, the burn-in period might be long since simulation inference is generally less precise from correlated draws than from the same number of independent draws. Any serial correlation after the burn-in period is not necessarily a problem since at convergence, the sample draws are identically distributed and thus for simulation inference we usually neglect the order of simulation draws. Other issues that affect convergence are bad starting parameter values and a multi-modal target distribution with some of its probabilities close to zero leading to poor mixing and local maxima problem. In general, using an optimal starting point close to the center of the chain is expected to speed up convergence. Moreover there exist formal approximations of calculating the effective number of independent draws needed from a particular simulation sequence (68). This approach is possible only in Gibbs samplers with multiple chains (68). Multiple chains starting at independent positions of the random variable space could help improve on coverage of parameter space and thus alleviate the problem of poorly mixed chains. In problems involving large number of parameters where computer storage is a problem, thinning the sequences by keeping every k^{th} simulation draw from each sequence and discarding the rest is an option. If the sequences have reached approximate convergence the thinned values can directly be used for parameter inference. Thinning also reduces the autocorrelation (see next subsection) within sequence samples.

3.2.6 Monitoring the convergence of each parameter of interest

We can never be sure if a chain in the Gibbs sampler has converged, but there are several tests we can do, both visual and statistical, to see if the chain of each parameter of interest(estimand) converged.

3.2.6.1 Trace plots

One way to see if our chain has converged is to see how well our chain is mixing, or moving around the parameter space. If our chain is taking a long time to move around the parameter space, then it will take longer to converge. A traceplot is a plot of the iteration number against the value of the draw of the estimand at each iteration. We can see whether our chain gets stuck in certain areas of the parameter space, which indicates bad mixing.

3.2.6.2 Autocorrelation

Another way to assess convergence is to assess the autocorrelations between the draws of our Markov chain for each estimand. The lag k autocorrelation ρ_k is the correlation between every draw and its k^{th} lag

$$\rho_k = \frac{\sum_{i=1}^{n-k}(y_i - \bar{y})(y_{i+k} - \bar{y})}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.11}$$

We would expect the k^{th} lag autocorrelation to be smaller as k increases (our 2^{nd} and 50^{th} draws should be less correlated than our 2^{nd} and 4^{th} draws). If autocorrelation is still relatively high for higher values of k, this indicates high degree of correlation between our draws and slow mixing.

3.2.6.3 Gelman and Rubin diagnostic

Gelman and Rubin diagnostics (88, 89) are based on analyzing multiple simulated chains by comparing the variances within each chain and the variance between chains. Large deviation between these two variances indicates nonconvergence. Suppose we have simulated m parallel sequences, each of length n after discarding the first half of the simulations. For each scalar estimand ω if we label the draws as ω_{ij} $(i = 1, ..., n; j = 1, ..., m)$, then the between- and within-sequence variances B and W can be calculated as:

$$\begin{aligned} B &= \frac{n}{m-1}\sum_{j=1}^{m}(\bar{\omega}_{.j} - \bar{\omega}_{..})^2, \text{ where } \bar{\omega}_{.j} = \frac{1}{n}\sum_{i=1}^{n}\omega_{ij}, \bar{\omega}_{..} = \frac{1}{m}\sum_{j=1}^{m}\omega_{.j} \\ W &= \frac{1}{m}\sum_{j=1}^{m}s_j^2, \text{ where } s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\omega_{ij} - \bar{\omega}_{.j})^2. \end{aligned} \tag{3.12}$$

Note that B cannot be calculated for only one sequence. We can estimate the $var(\omega|Data)$, the marginal posterior variance of the estimand, by a weighted average of W and B , given as:

$$\hat{var}^{+}(\omega|Data) = \frac{n-1}{n}W + \frac{1}{n}B. \tag{3.13}$$

The convergence of the Gibbs sampling is monitored by estimating the factor by which the scale of current distribution for ω might be reduced if the simulations were continued in the limit as $n \rightarrow \infty$. This important scale reduction measure also known as the Gelman and Rubin statistic is estimated as :

$$\hat{R} = \sqrt{\frac{\hat{var}^{+}(\omega|Data)}{W}} \tag{3.14}$$

which reduces to 1 as $n \rightarrow \infty$. A high $\hat{R}$ recommends that further simulations be made to improve on the target distribution for the estimand. To investigate convergence for the entire posterior distribution, the potential scale reduction factor is calculated for all scalar estimands. Upper and lower confidence limits can also be estimated to account for variability between chains. Approximate convergence is diagnosed when the upper limit is close to 1. The confidence limits are based on the assumption that the stationary distribution of the estimand under examination is normal. Hence transforming the scalar estimands to approximately normal may be useful.

3.2.6.4 Geweke diagnostic

The Geweke diagnostic(90) takes two nonoverlapping parts (usually the first 0.1 and last 0.5 proportions) of the Markov chain in the sampling period and compares the means of both parts, using a difference of means test to see if the two parts of the chain are from the same distribution (null hypothesis). The test statistic is a standard Z-score with the standard errors adjusted for autocorrelation. A large Z-score suggests possible convergence failure.

3.2.6.5 Raftery and Lewis diagnostic

Suppose we want to measure some posterior quantile of interest q and we want a diagnostic test that evaluates the accuracy of the estimated percentiles. The Raftery-Lewis test (91, 92) is designed for this purpose. If we define some acceptable tolerance

r for q and a probability p of being within that tolerance, the Raftery and Lewis diagnostic will calculate the number of iterations N and the number of burn-ins M necessary to satisfy the specified conditions. This diagnostic was designed to test the number of iterations and burn-in needed by first running and testing shorter pilot chain. In practice, we can also just test our normal chain to see if it satisfies the results that the diagnostic suggests. However this diagnostic measure will differ depending on which quantile q you choose and estimates tend to be conservative in the sense that it will suggest more iterations than necessary. Furthermore, it only tests marginal convergence on each parameter but nevertheless, it often works well with simple models.

3.2.6.6 Heidelberg and Welch diagnostic

The Heidelberg and Welch diagnostic (93, 94) calculates a test statistic (based on the Cramer-von Mises test statistic used for comparing two empirical distributions (95) to accept or reject the null hypothesis that the Markov chain is from a stationary distribution. The test is successively applied, firstly to the whole chain, then after discarding the first 10%, 20%, ... of the chain until either the null hypothesis is accepted, or 50% of the chain has been discarded. The latter outcome constitutes “failure” of the stationarity test and indicates that a longer Gibbs run is needed. If the stationarity test is passed, the number of iterations to keep and the number to discard are reported. More on this diagnostic can be read from the work of Heidelberger and Welch (1983)(94).

3.3 Motivation of Gibbs sampling in modeling the dynamics of a cell decision process

So far we have summarized all the procedures and building blocks involved in generating a Gibbs sampler. However, an efficient Gibbs sampler requires the appropriate specifications of the conditional distributions needed to generate the simulation samples. Of course this depends on the particular application and since this thesis is mainly about network reconstruction and parameter estimation in networks using Nested Effect Models(NEMs), the last section of this chapter motivates the use of Gibbs sampling in modeling the dynamics of a network or cell decision making within the framework of Nested Effects Models. Recall that in NEMs we have the silencing genes or knocked down genes called S-genes which form the core model and the effect genes, E-genes which correspond to the extended model. A complete model will be a network with edges linking E-genes to S-genes as well as edges between S-genes. A priori we don't know the E gene positions or which E-genes are regulated by which S-genes. These E-S-gene positions form unknown parameters in the model. In addition signaling networks are made up by collections of interacting signaling pathways which can activate or inhibit gene expression in the cell. These can be represented as unknown discrete edge weight parameters between the S-genes. Also the regulatory network of a biological process of an organism is highly dynamic with different sections of the network actively used under different conditions or over a period of time(96). To understand the network dynamics, a third set of parameters corresponding to unknown signal propagation rates or time delays between the S-genes could be added to the model. Learning both the network topology and the network dynamics involves estimating the joint distribution of a given network model and its associated parameters. With so many parameters involved especially if the network is large, the joint distribution would be difficult to compute analytically. Gibbs sampling would be a useful alternative to estimate the marginal distributions of all model parameters without necessarily estimating the joint distribution. The next chapter shows the implementation of Gibbs sampling for parameter estimation within the framework of Nested Effects Models.

4

Dynamic Nested Effects Models

In the introduction, I motivated the need for inferring the dynamics of regulatory networks in a slow going process such as cell differentiation. In the next sections, I develop a new Bayesian method known as the Dynamic Nested Effects models (DNEMs). This approach is an extension to NEMs introduced in chapter 2 to infer the dynamics of a given network which is an important limitation in NEMs. I first introduce a method for estimating parameters in a given network based on perturbation time series data using Gibbs sampling. I present an algorithm to infer signal propagation rates in a given network with particular application to transcriptional signaling in stem cell differentiation.

4.1 Dynamic Nested Effects Models(DNEMs)

Time delays between signaling events cannot be observed directly. They need to be estimated. In practice we observe signal propagation times from some intervention say S_1 to some target genes read outs(Figure 4.1). We don't observe the time delays between S_1 and S_2. We would like to estimate the rate of signal propagation from S_1 and S_2. In general, the challenge is that given the hierarchy of signaling steps we want to estimate the signal propagation rates for all edges given interventional data. Of course in order to do so we first need to estimate the hierarchy of signaling genes, identify which gene expression profiles are connected to which steps in the hierarchy and finally estimate all the signal propagation rates. DNEMs address these problems.

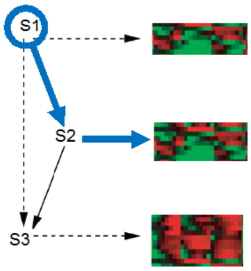
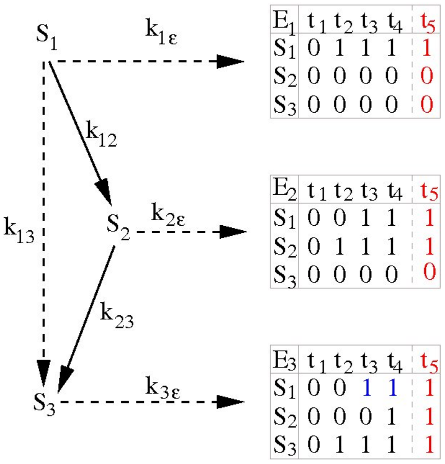


Figure 4.1: Time delays in a given network - Time delays between signaling events e.g. delay from S_1 to S_2 cannot be observed directly. They need to be estimated. We only observe the time after intervention at say S_2 to some readout gene expression.

Figure 4.2 illustrates a simplified and easy to understand solution to the problem which is basically the idea of DNEMs. The graph on the left of the tables is a transitively closed graph on 3 Signaling genes(S_1, S_2, S_3). The tables give the time series binary data of effects for all target genes (E_1, E_2, E_3) after intervention on all signaling genes. A one indicates the signal has already reached the target gene by time t_j, while zero indicates that the expression of this gene has not yet changed or no interventional effect has occurred. Looking at the last time point t_5 one sees the accumulation of effects for all target genes forming a nested structure of effects which is in conformity with the hierarchy of the graph topology. Signals starting in S_1 reach E_2 one time unit after they have arrived at E_1 suggesting that signal propagation from S_1 to S_2 takes one unit of time. The same argument using the data from perturbation of S_2 suggests that it takes two time units to propagate from S_2 to S_3. Consequently, going from S_1 to S_3 *via* S_2 takes 3 time units. However, the time delay from perturbation of S_1 to observing effects in E_3 is only 1 time unit (marked in blue). This suggests the existence of a direct signal flow from S_1 to S_3. Evidence comes from the two blue ones. In case they were zeros, the time delay between S_1 and S_3 would have been the sum of times spent when going via S_2. In this case, there would be no evidence for a shortcut pathway and we would decide on the more parsimonious graph. Furthermore, the existence of a direct path combining with that of the indirect path gives evidence of the presence a Feed-Forward Loop. Thus we can use estimated time delays to demonstrate the existence of FFLs. A real world analysis is more difficult than the toy example. Signal propagation is a stochastic process, measurements are prone to noise, and we do not know which E-genes are controlled by which S-genes. These sources of uncertainty are addressed by DNEMs.



E_1	t_1	t_2	t_3	t_4	t_5
S_1	0	1	1	1	1
S_2	0	0	0	0	0
S_3	0	0	0	0	0

E_2	t_1	t_2	t_3	t_4	t_5
S_1	0	0	1	1	1
S_2	0	1	1	1	1
S_3	0	0	0	0	0

E_3	t_1	t_2	t_3	t_4	t_5
S_1	0	0	1	1	1
S_2	0	0	0	1	1
S_3	0	1	1	1	1

Figure 4.2: Idea of DNEMs in an elementary example - Shown is the hierarchical structure of a network and discrete time series data for three E-genes. A one indicates that a signal has reached the E-gene, while a zero indicates that the expression of this gene has not yet changed. Note, that the graph topology is consistent with the nested structure of ones in the final time point t_5, shown in red.

We model signaling as a stochastic process with exponentially distributed time delays. Given a hierarchy of signaling steps, DNEM assumes exponential time delays between signaling steps. The rate constants of the exponential distributions differ from case to case and are the main parameters of the model. All edges of a transitively closed network are associated with an individual rate constant, whose posterior distribution is inferred using Gibbs sampling. Since there are possibly several decision making steps between and input signaling gene and an output signal we deal with convolutions of exponentials. Furthermore, we assume that if a S-gene has multiple incoming edges, the first blocked signal blocks activation. In other words we assume underlying AND gates for the S-gene interactions. As explained in the introduction of this thesis, molecular signaling in the cytoplasm occurs at high rates, direct signal propagation via transcription and translation at intermediate rates, and secondary effects at low rates. The joint posterior of the rate constants will be used to analyze the interplay of signaling networks and gene expression. It is also used to unravel molecular signal flow in cells.

4.1.1 DNEM algorithm

The input of a DNEM consists of (a) a set of microarray time series that measure the response of cells to molecular perturbations, and (b) a transitively closed directed acyclic graph on vertex set $\mathbf{S}$ representing a hypothetical hierarchical structure of upstream/downstream relations. This graph can be derived from any of the methods outlined in chapter 2 or from literature. The output consists of (a) the joint posterior distribution of rate constants describing the dynamics of signal propagation, and (b) a not necessarily transitive subgraph of the input graph that describes signal flow rather than upstream/downstream relation.

4.1.1.1 Model parameters

Let $D(i, k, l, s)$ denote the expression measurement of E_k in time point t_s of the l'th replication of a time series recorded after perturbation of S_i. Following Markowetz *et al.*(2007) (57), we assume that the data is binary, indicating whether interruption of signal flow was observed at a particular E-gene at a particular time point. A zero encodes the wild type expression level of a gene, while one encodes that the expression of this E-gene changed due to perturbation induced signal propagation. Later on we consider the case of continuous read outs. We assume that the time spent for propagating a signal from node S_i to node S_j is exponentially distributed with a rate constant k_{ij}. Note that the expected time spent in this step of signal transduction is $1/k_{ij}$. Fast processes are associated with high rate constants, while slow processes are associated with small rate constants. Exponential distributions are widely used to model temporal processes in complex systems (97, 98). Recall that we do not observe the time spent for signal propagation between S-genes directly. Instead, we observe the time delay between a perturbation of an S-gene and the occurrence of downstream effects in E-genes. Following Markowetz *et al.* (57) we introduce parameters $\mathbf{\Theta} = (\theta_1, \ldots, \theta_N)$ to link E- to S-genes. If $\theta_k = i$, then E_k is linked to S_i. Moreover, we assume that every E-gene is linked to a single S-gene. The set of E-genes attached to the same S-gene is a regulatory module under the common regulatory control of the S-gene. The module of E-genes attached to S_i is denoted by $\mathcal{E}_i$. Finally, we introduce additional rate constants $k_{i\mathcal{E}}$ that represent the time delay between activation of S_i and regulation of its target module $\mathcal{E}_i$ Figure 4.2. A single common rate is used for all E-genes in the module.

Similar to ideas from Tresch and Markowetz (59), we add an additional node denoted by $\bigoplus$, which is not connected to any of the S-genes. However, E-genes can be linked to this node, if they do not fit in any of the $\mathcal{E}_i$. The $\bigoplus$-node implicitly selects E-genes. Genes linked to $\bigoplus$ are excluded from the model. Figure 4.3 gives a complete model parameterization for 3 S-genes with all E-gene position and rate parameters.

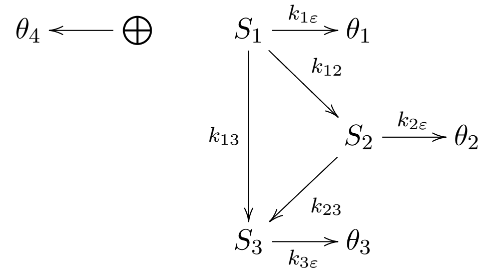


Figure 4.3: Parameterization of DNEM- There are two sets of parameters involved in the DNEM. The E-gene positions(θ) and the rate parameters (k) for signal propagations

We denote the complete set of rate constants including rates between S-genes and rates between S- and E-genes by $\mathbf{K}$. A priori, we do not know which E-genes fall into which modules. The joint posterior distribution of $\boldsymbol{\Theta}$ and $\mathbf{K}$ will be inferred from the data. While the θ_k are discrete parameters by nature, rate constants are usually modelled as continuous parameters. However, for the sake of computational efficiency, we confine the rates to a discrete set of values denoted by $(\kappa_0, \dots \kappa_T)$. If the data includes time points $(t_1, \dots, t_T)$, we choose $(\kappa_0, 1/t_1, \dots, 1/t_T)$, where κ_0 is set to a high value (i.e. 1,000) that represents the very fast signal transduction through post translational protein modification like phosphorylation. Overall, we have a set of discrete parameters only $(\mathbf{K}, \boldsymbol{\Theta})$.

4.1.1.2 Prior distributions for model parameters

Assuming independent prior distributions for $\mathbf{K}$ and $\boldsymbol{\Theta}$, Bayes's theorem yields

$$P(\Theta, K|D) = \frac{P(D|K,\Theta)P(K)P(\Theta)}{P(D).}$$

The prior distribution $P(\boldsymbol{\Theta})$ can be chosen to incorporate prior knowledge on the interactions of S- with E-genes. Such information might be derived from ChIP data or regulatory motif analysis. The prior provides an interface, through which the model can be linked to different biological data types in integrative modeling approaches. Here we use the prior for calibrating E-gene selection. We set $p(\theta_k = \bigoplus)$ to Δ, while distributing the remaining weight of $1 - \Delta$ uniformly on the values $1, \dots, n$.

Similarly, the prior distribution $P(K)$ yields an interface for incorporating biological knowledge. If one knows that S_1 and S_2 fall into the same molecular signaling pathway, one can set $P(k_{12} = \kappa_0)$ to one, because signaling will operate on a high rate. In this thesis we exploit the fact that transcription takes hours and set $P(k_{i\mathcal{E}} = \kappa_0)$ to zero, while assuming a uniform prior for the remaining values. It is also possible to model the rate parameter as a continuous variable. In this setting, the unknown time delays are assumed to follow an exponential distribution

$$T_u \sim k_u \exp(-k_u \tau)$$

and we assume that the rate constants follow a conjugate gamma prior distribution

$$k_u \sim \text{Gamma}(k_u, \alpha_u^{'}, \beta_u^{'}) = \frac{\beta_u^{' \, \alpha_u^{'}}}{\Gamma(\alpha_u^{'})} k_u^{\alpha_u^{'}-1} \exp(-k_u \beta_u^{'})$$

with $\alpha_u^{'} > 0$ shape and $\beta_u^{'} > 0$ scale parameters respectively. Assuming independent priors for the time delays, the posterior will again be gamma distributed and the density of signal propagation between and input and output signal will be some form of convolution of gamma distributions. Closed form expressions for convolution of independent gamma random variables have already been established (99).

4.1.1.3 Probability density of signal propagation between input and output signal

Let us first consider a fixed linear path g in Φ, which connects the S-gene S_i with the E-gene E_k:

$$S_i \xrightarrow{k_1} S_{j_1} \cdots \xrightarrow{k_{q-1}} S_{j_{q-1}} \xrightarrow{k_q} E_k,$$

where for simplicity of notation we reduce the double indices of rate constants to single indices and write $k_1, k_2, \dots, k_q$ to denote the rate constants. We are interested in the time needed for propagating a signal from S_i down the path to E_k. More precisely, we want to calculate the probability, that the signal has reached E_k before some fixed time point t^*. If Z_g is the sum of q independent, and exponentially distributed random variables with rate constants $k_1, \dots, k_q$, then this probability equals $P(Z_g < t^*)$. The density function of Z_g is given by the convolution of independent exponential distributions

$$\Psi(t)_g = \int_0^\infty \cdots \int_0^\infty \delta\left(t - \sum_{u=1}^{q} \tau_u\right) \prod_{u=1}^{q} \psi_u(\tau_u)\, d\tau_1 \ldots d\tau_q,$$

where $\psi_u(\tau) = k_u \exp(-k_u \tau)$ is the density of an exponential with rate k_u. Laplace-transformation yields a closed form for the cumulative distribution function of Z_g

$$F_g(t) = \sum_{b=1}^{q} \prod_{a \neq b} \left\{ \frac{k_a}{k_a - k_b} \right\} [1 - \exp(-tk_b)]\,. \tag{4.1}$$

See Appendix for a complete proof. Note that the right hand side is not defined if two or more of the k_u are identical. However, as right and left limits exist and are identical, we can evaluate the probability by adding tiny distinct jitter values to the k_u. However an exact function for the convolution of exponentials for a general K_u has been established by Jasiulewicz and Kordecki (2003)(100).

4.1.1.4 Probability density of signal times generalized to phase-type distributions

It is also possible to consider Equation 4.1 as a special case of the phase-type distribution. The phase-type distribution is the time to absorption of a finite state Markov process. If we have a $u + 1$ state process, where the first u states are transient and the state $u + 1$ is an absorbing state, then the distribution of time from the start of the process until the absorbing state is reached is phase-type distributed. In the context of DNEM, we could consider signal flow between S-genes in a particular path as transient states and the signal flow from the last but one S-gene to the target E-gene as the absorption state. If we assume exponential time delays between edges, this becomes the hypoexponential(*Hypo*) (101) if we start signal propagation from an input S-gene and move skip-free from S-gene$_i$ to S-gene$_{i+1}$ with rate k_i until S-gene$_u$ transitions with rate k_u to the target E-gene$_{u+1}$. This can be written in the form of a subgenerator matrix,

$$K = \begin{pmatrix} -k_1 & k_1 & 0 & \cdots & 0 & 0 \\ 0 & -k_2 & k_2 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & -k_{u-2} & k_{u-2} & 0 \\ 0 & 0 & \cdots & 0 & -k_{u-1} & k_{u-1} \\ 0 & 0 & \cdots & 0 & 0 & k_u \end{pmatrix}$$

Keeping our notation consistent we denote the above matrix $\mathbf{K}\equiv \mathbf{K}(k_1,\ldots,k_u)$. If the probability of starting in each of the u states is $\boldsymbol{\alpha} = (1,0,\ldots,0)$ then $Hypo(k_1,\ldots,k_u) = PH(\boldsymbol{\alpha},\mathbf{K})$. This distribution can be characterized as follows:

A random variable $X \sim Hypo(k_1,\ldots,k_u)$ has cumulative density function (cdf)

$$F(x) = \mathbf{1} - \boldsymbol{\alpha}\exp^{x\mathbf{K}}\mathbf{1}$$

where $\mathbf{1}$ is a column vector of ones of size u and $\exp^A$ is the matrix exponential of A. When $k_i \neq k_j$ for all $i \neq j$, the cdf becomes Equation 4.1 with moment generating function given as:

$$E(X^n) = (-\mathbf{1})^n n!\boldsymbol{\alpha}\mathbf{K}^{-n}\mathbf{1}$$

The characterization of phase-type distributions in this form makes it easier to estimate the distributions of certain features of stochastic networks such as the distribution of length of shortest paths between nodes (102).

4.1.1.5 Probability density function when alternative paths do not share edges

In the general case a signal can be propagated from S_i to E_k *via* multiple alternative paths. In this case we assume that the fastest path determines the time delay for downstream effects to be seen. We enumerate all linear paths connecting S_i to E_k. We introduce an algorithm on how we enumerate all paths in a directed graph between two nodes in the next chapter. For each path we construct a random variable Z_u as described above. If the alternative paths do not share edges(independent signals), the probability that the signal has arrived at E_k before time t^* *via* at least one of the paths is given by

$$\begin{aligned}
P_{S_i\rightarrow E_k}(t^*) &= P(W = min(Z_1,...,Z_n) \leq z) \\
&= 1 - P(\bigcap_{u=1}^{n}(Z_u > z)), \text{ since all } Z_u \text{ have the same distribution.} \\
&= 1 - P(Z_1 > z)\cdots P(Z_n > z) \\
&= 1 - [(1 - P(Z_1 \leq z))\cdots(1 - P(Z_n \leq z))] \\
&= 1 - \prod_u (1 - F_u(t^*)) \qquad (4.2)
\end{aligned}$$

4.1.1.6 Probability density function when alternative paths share edges

Equation (4.1) gives a closed formula for the delay distribution for the signaling along a linear path. However there is no closed expression describing the delay distribution for the signaling from an S-gene to an E-gene in a directed acyclic graph. Therefore we need to resort to sampling techniques. We illustrate this with a simple example. The figure below correspond to a signaling graph with 5 S-genes ($S_1, ..., S_5$) with exponentially distributed time delays with rate constants $(k_1, ..., k_5)$ on the edges. Notice the edge between S_4 and S_5 occur in the two alternating paths between S_1 and S_5. We want to estimate the signal propagation density from say S_1 to S_5 which is a minimum of two dependent random variables.

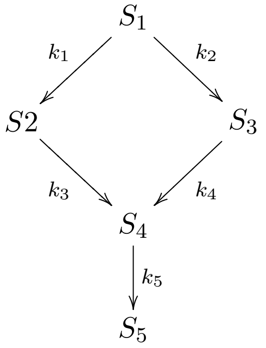


Figure 4.4: Graph of 5 S-genes with alternative paths sharing an edge- The signal propagation rates from exponential time delays are represented on the edges.

The procedure is as follows: For a fixed time lag, τ, we draw independent identically distributed exponentials with rate parameters $(k_1, ..., k_5)$. Usually about 10000 samples. If we define the random variables, V_1 = Sum of exponentials corresponding to edges with rates constants (k_1, k_3, k_5) and V_2 = Sum of exponentials corresponding to edges with rates constants (k_2, k_4, k_5). Then the minimum of (V_1, V_2) is calculated for the 10000 independent draws. The cumulative density function for the distribution for $T_{min(V_1,V_2)}$ can be estimated using the updated simulated draws. This algorithm can be generalized for any given network with signaling rate constants on the edges. In practice this approach will make our algorithm very time consuming and unrealistic due to the long running times of the Gibbs sampler. However we will show from both simulated and real studies that we can generally approximate the distribution for dependent signals with that of independent signals without much loss of information. This approximation is based on the assumption that the interactions among merging pathways can be neglected similar to the mean-field approximation from many body theories in statistical physics. Equation 4.2 becomes

$$P_{S_i \to E_k}(t^*) \approx 1 - \prod_u (1 - F_u(t^*)) \tag{4.3}$$

4.1.1.7 Sensitivity analysis between independent and dependent signals

Let us consider the graph in Figure 4.4 as our signaling network with rate constants $(k_1 = 1, k_2 = 1/2, k_3 = 1/3, k_4 = 1/4, k_5 = \kappa)$, $\kappa \in \{5, 1, 0.8, 0.7, 0.6, 0.5, 0.3, 0.2, 0.1\}$. Simulated distributions for signaling between nodes S_1 and S_5 in the graph under independent and dependent alternating paths are compared by the QQ-plots in Figure 4.5. We observe most of the points lie on the line through the origin for larger rate constants indicating that the underlying distributions are similar. The offset between the line and the points is negligible for smaller data points and gradually increases towards the tail with the largest offset between the distributions for the paths between S_1 and S_5 for rate $k_5 = 0.1$. The plots demonstrate that using Equation 4.3 for the cdf for signaling between dependent paths underestimates the time delays in general.

4.1.1.8 Marginal likelihood for discrete models

Equations (4.1) and (4.2) describe the stochastic nature of signal propagation in the cell. Before calculating the likelihood, we need to consider a second source of stochasticity, namely measurement error. Following Markowetz *et al.*(2007) (57), we denote the probabilities for false positive and false negative signals by α and β respectively (Table 2.1). Assuming conditional independence, the likelihood factorizes into

$$\begin{aligned} P(D|K,\Theta) &= \prod_{D=1} P_{S_i \to E_k}(t_s)(1-\beta) + (1 - P_{S_i \to E_k}(t_s))\alpha \\ &\times \prod_{D=0} P_{S_i \to E_k}(t_s)\beta + (1 - P_{S_i \to E_k}(t_s))(1-\alpha), \end{aligned} \tag{4.4}$$

where the first product is over all data points, for which we observe a downstream effect, and the second product over those for which we do not. Observations from E-genes linked to the $\bigoplus$-node generate neutral likelihood values of 0.5 independent of all other parameters.

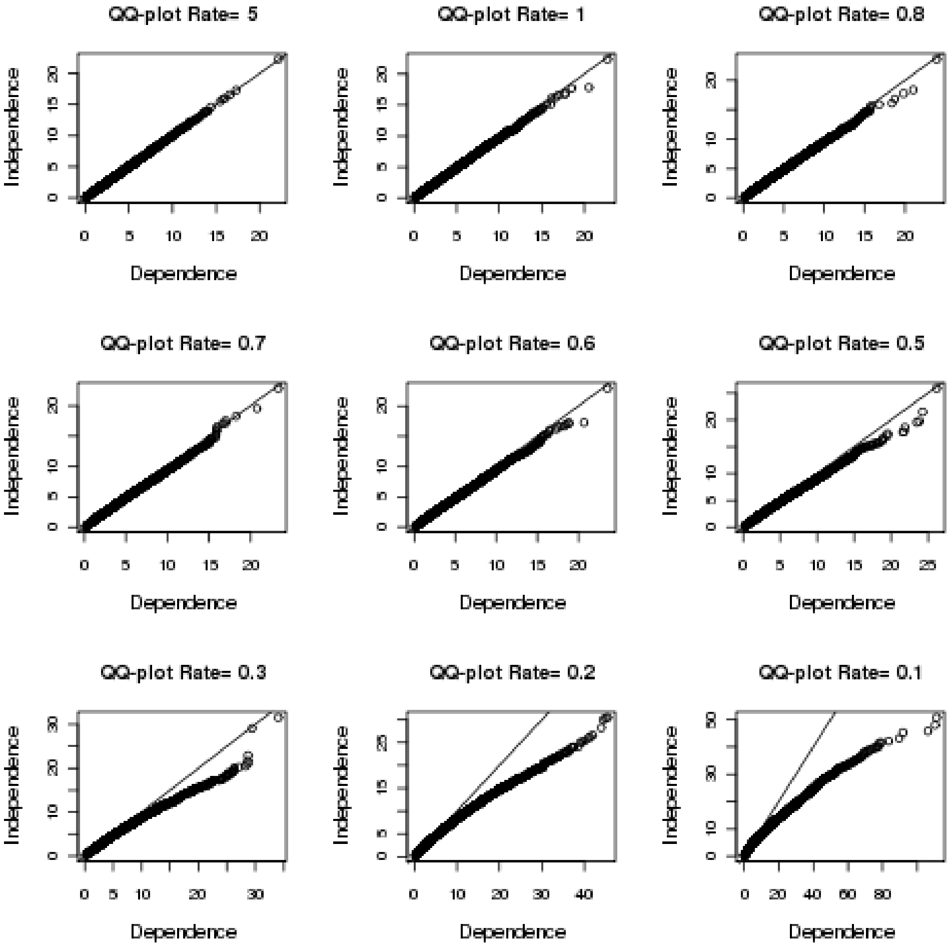


Figure 4.5: **QQ-plots comparing distributions of signal times between two nodes S_1 and S_5 in the graph in Figure 4.4 under dependent and independent assumptions.** - The offset between the line and the points is negligible for smaller data points and gradually increases towards the tail. The QQ-plot for the distributions between S_1 and S_5 show an underestimation of time delays.

4.1.1.9 Marginal likelihood for continuous models

NEMs in (59, 103) circumvent the use of α and β in the likelihood by using the information on the probability of a gene being differentially expressed. With respect to microarray data with replicates at time points these probabilities can be easily estimated using linear models(104). Assuming we have these probabilities Equation 4.4 becomes

$$P(D|K,\Theta) \;=\; \prod_D P_{S_i \to E_k}(t_s) p_{ikls} + (1 - P_{S_i \to E_k}(t_s))(1 - p_{ikls}), \tag{4.5}$$

where p_{ikls} is the probability of gene E_k being differentially expressed in time point t_s for experiment replication l after perturbation S_i. Equation 4.5 is just the product over all data points.

4.1.1.10 Discrete Gibbs sampling

With N E-genes, n S-genes and L edges in the input graph, the model comprises $N+n+L$ discrete parameters. For simplicity of notation, we reduce the double indices of rate constants to single indices such that the joint posterior is written

$$P(k_1, \ldots, k_{L+n}, \theta_1, \ldots \theta_N | D).$$

We initialize the parameters with random values from their domains. Then we iteratively cycle through all rate constants updating them by sampling from the conditional posterior distributions

$$p(k_i | \mathbf{K} - \{k_i\}, \boldsymbol{\Theta}, D).$$

With only discrete parameters, updating is straight forward: We calculate all values

$$p(k_i = \kappa_j)\, p(D | \mathbf{K} - k_i, \boldsymbol{\Theta}, k_i = \kappa_j),$$

normalize them to sum up to one, and draw a new value for k_i from this distribution. The iteration is completed by similarly updating all θ_k. We sample 10,000 times from the joint posterior distribution of parameters, discard the first 1,000 draws as burn in time, and summarize the remaining ones for inference of signal propagation. Choosing suitable values for the tuning parameters α and β protects the conditional posterior distributions from singularity, and ensures good mixing properties of the Gibbs sampler.

4.1.1.11 Inference of E-gene positions

Recall that we do not know which E-genes are controlled by which S-genes. In order to update the rate parameters in the Gibbs sampler, we first need to attach the E-genes to their rightful positions. One way is to estimate these positions from the last time point corresponding to the accumulation of effects representing the nested structure in the data and then use this as fix parameters in the Gibbs. In this situation we only update the rate parameters making our algorithm faster. Given a silencing scheme Φ, the posterior probability for an edge between and S-gene S_j and an E-gene E_i is given by

$$P_{\alpha,\beta}(\theta_i = j|\Phi, D_T) \quad = \quad \frac{P_{\alpha,\beta}(\theta_i = j)}{P(D_T)} \prod_{k=1}^{l} p_{\alpha,\beta}(e_{ik}|\Phi, \theta_i = j) \tag{4.6}$$

where D_T is the data matrix at the last time point T. The prior $P_{\alpha,\beta}(\theta_i = j)$ can be non-informative such as a uniform distribution although in general, the prior could take any other form as long as it is the same form as in the computation of the marginal likelihood in Equation 2.6. The E-genes attached with high probability to an S-gene are interpreted as a regulatory module, which is under the common control of the S-gene. Alternatively, we can sample the E-gene positions directly from their conditional posterior distributions inside the Gibbs Sampler and use this sample to update the rate parameters. For the purpose of illustration we focus only on a cascade with two rate parameters k_1, k_2 and 1 E-gene (Figure 4.6).

We initialize all parameters randomly, and at each iteration say t', we update all the E-gene positions as follows. For a given E-gene say E_k we attach to S_1 and calculate the posterior probability for attaching the E-gene to S_1 given the model. Similarly we attach it to S_2, S_3 and $\bigoplus$ respectively to get the complete distribution for attaching E_k to an S-gene or not. We normalize the distribution to sum to one and draw a E-gene position. Once we have a new set of E-gene positions we then update all Ks in a similar manner by sampling from their conditional posterior distributions. This completes one iteration step. After several iterations, inference on E-gene positions can be derived from their posterior samples using MAP.

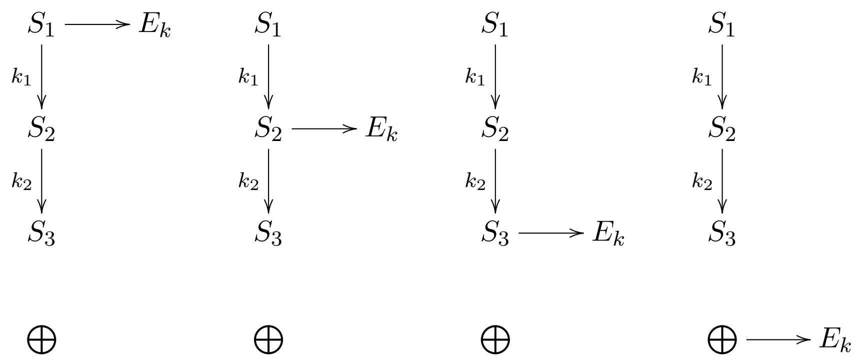


Figure 4.6: Updating E-gene positions inside Gibbs sampling- For a given E-gene say E_k we attach to S_1 and calculate the posterior probability for attaching the E-gene to S_1 given the model. Similarly we attach it to S_2, S_3 and $\bigoplus$ respectively to get the complete distribution for attaching E_k to an S-gene. We normalize the distribution to sum to one and sample a new E-gene position.

4.1.1.12 Inference of signal flow

Under the natural assumption that perturbation effects propagate down the signaling network to all descendants of a perturbed gene, the nested structure of downstream effects resolves the network only up to its transitivity class. Network topologies with identical transitive closures produce the same nesting of downstream effects and, hence, can not be distinguished. Temporal data hold the potential of further resolving these transitivity classes. DNEM starts from a transitively closed network. Posterior distributions are calculated across a discrete set of rate constants including a very small rate constant κ_{T+1}. As explained above, $k_{ij=\kappa_{T+1}}$ reflects network constellation, in which no signal is flowing through the edge from S_i to S_j. Note that if a rate constant is set to κ_{T+1}, the corresponding edge is not contributing to the likelihood according to Equation 4.2. The edge is effectively excluded from the model. Hence, in addition to estimating average time delays the Gibbs sampling procedure facilitates network refinement. If the posterior probability of the edge from S_i to S_j is $P[k_{ij=\kappa_{T+1}}|D] > p^*$, $p^* > 0.5$, we exclude the edge from the network. Of course the choice of p^* is subjective.

4.1.1.13 Model comparisons using DNEMs

Due to the long running times of the Gibbs sampler it is not possible to reconstruct the network topology from scratch as was done for standard NEMs in (57, 59, 103) through exhaustive search or greedy hill climbing. Nevertheless, we can discriminate between small numbers of candidate topologies using posterior odds for model comparison. Let us assume we have two hypothetical network topologies Φ_1 and Φ_2. The ratio of their posterior probabilities equals

$$\begin{aligned}\frac{P(\Phi_1|D)}{P(\Phi_2|D)} &= \frac{P(\Phi_1)}{P(\Phi_2)} \times \text{Bayes factor}(\Phi_1;\Phi_2) \qquad (4.7)\\ \text{where Bayes factor}(\Phi_1;\Phi_2) &= \frac{P(D|\Phi_1)}{P(D|\Phi_2)} \qquad (4.8)\\ &= \frac{\int\int P(\Theta_1,K_1|\Phi_1)P(D|\Theta_1,K_1,\Phi_1)d\Theta_1\,dK_1}{\int\int P(\Theta_2,K_2|\Phi_2)P(D|\Theta_2,K_2\Phi_2)d\Theta_2\,dK_2} \qquad (4.9)\end{aligned}$$

with Θ_i and K_i representing the parameters in model Φ_i. The Bayesian model comparison does not depend on specific parameter settings. Instead, it considers the probability of the model considering all possible parameter values. An advantage of using the Bayes factor is that it guards against overfitting by automatically, and quite naturally, including a penalty for including too many degrees of freedom. The integrals in the Bayes factor can be approximated by averages along the Gibbs sampling trajectories. In practice, this is not feasible due to the numerical representation of the tiny likelihood values. We therefore look at another approximate approach which measures the distance of data to each of the models. In this situation, even if none (or all) of the models fit the data, it can be informative to compare their relative fit. Here, we use the *deviance information criterion* (DIC) of Spiegelhalter *et al.*(2002) (105) given as:

$$\text{DIC} = \hat{V}^{pred}_{avg}(D) = 2\hat{V}_{avg}(D) - V_{(\hat{\Theta},\hat{K})}(D)$$

with $V_{(\hat{\Theta},\hat{K})}(D)$ defined as $-2\text{log}p(D|\hat{\Theta},\hat{K})$ corresponding to the deviance which gives a summary of the discrepancy between the data and model and depends only on D. One can use MAP estimates for $(\hat{\Theta},\hat{K})$.

$$\hat{V}_{avg}(D) = \frac{1}{L}\sum_{l=1}^{L} V(D,(\Theta^l,K^l))$$

averages the discrepancy $V(D,(\Theta,K))$ over the posterior distribution. The estimated average discrepancy is a better summary for model error than the deviance $V_{(\hat{\Theta},\hat{K})}(D)$.

In practice, the model Φ_i with the lowest DIC has the lowest estimated expected predictive error.

4.1.2 Speed up by stochasticity effects in signaling networks

Equation 4.2 is about calculating the distribution of a minimum of independent random variables $Y = min(X_1, ..., X_n)$, n finite. In general the expectation of this minimum is smaller than the minimum of expectations of the X_is since

$$\begin{aligned} E(Y) = E(min(X_1, ..., X_n)) &\leq E(X_i), \text{ for all } i \\ &\leq min(E(X_1), ..., E(X_n)). \end{aligned} \tag{4.10}$$

Equation 4.2 describes the stochastic nature of signal propagation in the cell. From Figure 4.5 and based on Equation 4.2 the average overall time delay between S_i and E_k is smaller than the average time delay associated with the fastest path connecting them because, with some positive probability, the in average slower process will be the actually faster one. We call this effect "**speed up by stochasticity**". This speed up by stochasticity effect is a consequence of the stochastic nature of time delays. We demonstrate this on the same example in Figure 4.4. We assume distinct signaling rates in this case between the edges as before but a fixed rate for $k_5 = 1/5$. The box plots in Figure 4.7 show that the expected minimum time delay between S_1 and S_5 ($E(Y)$) under both independent and dependent signaling is smaller than the minimum of the expected time delays $E(S_1S_2S_4S_5)$ and $E(S_1S_3S_4S_5)$ as expected. The difference between $E(Y)$ and $min(E(X_i))$ is biggest for independent X_i. It becomes somewhat less pronounced for dependent X_i. Hence the approximation in Equation 4.3 leads to an underestimation of time delays. Thats a systematic bias. If we use a model in which we estimate time delays instead of rate constants we do not have this problem. Then we can just use the minimum of estimated time delays. There is no speed up by stochasticity in this case. This will most likely lead to an overestimation of time delays, if the processes in reality have a stochastic nature. We investigate both scenarios using real application in the next section. We carryout a first test of our algorithm in simulation scenarios where data is artificially generated according to the model assumption. Finally we apply the DNEM algorithm to a data set on molecular mechanisms of self-renewal in murine embryonic stem cells to investigate the dynamics between 6 ESC transcription factors during early stage differentiation.

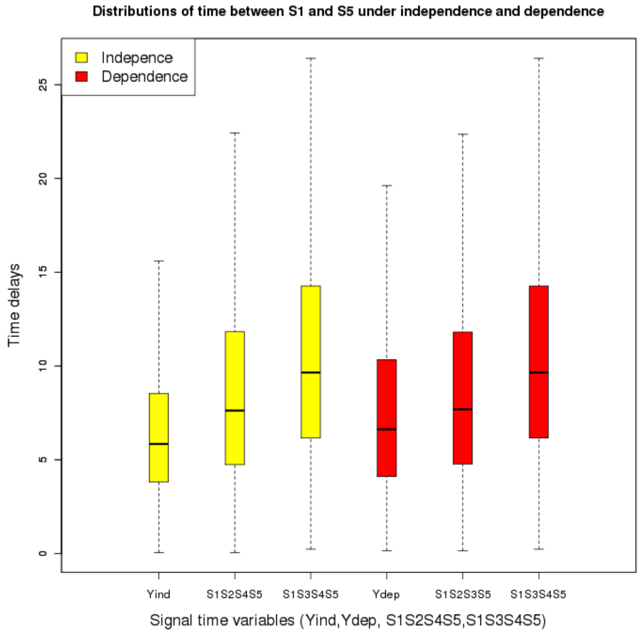


Figure 4.7: Speed up by stochasticity effect of DNEMs - Speed up by stochasticity effect is a consequence of the stochastic nature of time delays. Box plots show the distributions of time under independent and dependent signaling between paths $S_1S_2S_4S_5$ and $S_1S_3S_4S_5$ and the distribution of minimum time between S_1 and S_5 denoted as Y_{ind} and Y_{dep} for the independent and dependent conditions. The average overall time delay between S_1 and S_5 is smaller than the average time delay associated with the faster path connecting these nodes under both dependent(red) and independent(yellow) signaling.

4.2 Simulation results

A first test of validity of a complex data model is to test its performance in simulation scenarios where data is artificially generated according to the model assumption. Here, we show that our model recovers time delays in noisy data and detects transitive shortcut edges even in situations where time delay differences are subtle.

4.2.1 Data generation

We evaluate our method in the context of simulated data from the network shown in Figure 4.8. The topology of this network is identical to the one we derived from our analysis of early stem cell differentiation using static NEM. Note that the network is transitively closed. Time delays for signal propagation between *S*- and *E-genes* are set to 1. For signal propagation between *S-genes* we distinguish between transitive and non-transitive edges. While time delays along non-transitive edges are set to 3 in all simulation experiments, the time delays used for transitive edges varies across experiments, including the use of very high time delays (100) to simulate a network with virtually no signal flow through transitive edges. For all *E-gene* positions the expected data pattern across time points and perturbation experiments is calculated and artificial *E-gene* data is simulated by adding independent binary noise to these patterns using a range of different noise levels: $\alpha = 0.0, 0.1, 0.2, 0.3$ and $\beta = 1/2\,\alpha$. We simulate data for 20 *E-genes* per *S-gene* and one measurement per time point, resulting in a data array of 840 binary values. DNEM is run on this data using two independent runs of 5,000 iterations, from which the first 500 are discarded as burn in time, leaving 9,000 posterior samples per simulation. For calculating the likelihood, we set the tuning parameters α and β at 0.2 and 0.1, respectively. Note that with one exception, these values are different from those used in data generation.

4.2.2 A sparse network

In a first simulation we examine a sparse network without shortcut pathways. We generated data from the reference network in the absence of any transitive edge. However, when running DNEM on this data we included the transitive edges in order to validate that DNEM can accurately detect that the edges did not exist.

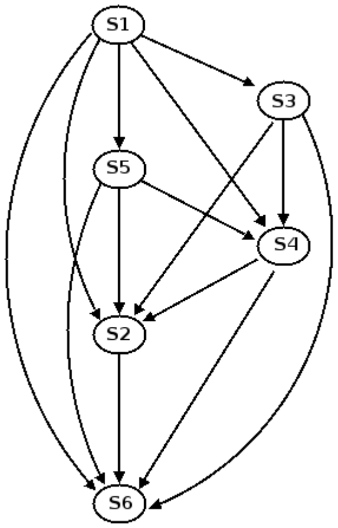


Figure 4.8: The transitive network used for simulation - The topology of this network is identical to the one we derived from our analysis of early stem cell differentiation using static NEM.

Figure 4.9(A-D) shows estimated average time delays (reciprocal rate constants) along the Gibbs sampling trajectories in the form of gray-scale intensity profiles. Light gray indicates high marginal posterior probability, while dark gray stands for low marginal posterior probabilities. The original time delays used in data generation are shown to the left of the heatmap, where an "x" indicates that the edge was excluded during data generation. We observe that posterior modes are generally close to true values. For high noise levels, the marginal posterior distribution are more disperse, nevertheless posterior modes are still close to their target values. More importantly, for the transitive edges that did not exist in data generation, we observe posterior distributions that concentrate weight on the "x" state. In this way, they hardly contribute to the likelihood, which is driven by shortest paths from S- to *E-genes*. Non-existing transitive edges (marked in red) automatically exit the model and do not interfere with the estimation of average time delays for the remaining edges. Using the cutoff $P(k_i = \kappa_{T+1}) > 0.6$ for the exclusion of an edge, we correctly exclude all non-existing edges for all noise levels.

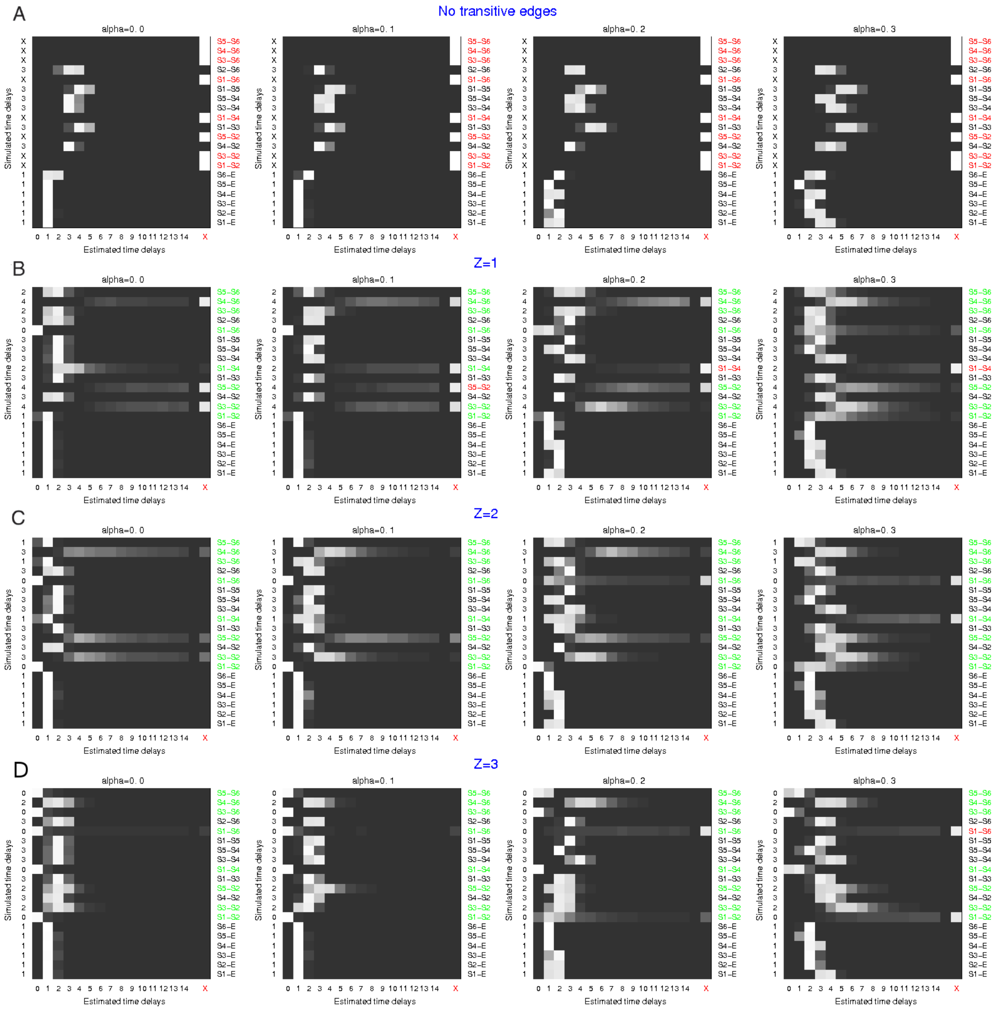


Figure 4.9: Heat map of the posterior distribution of average time delays - Rows correspond to edges of the network including those between S- and E-genes, while columns refer to average time delays. Marginal posterior probabilities are gray-scale colored with light gray indicating high and dark gray indicating low probability. Edges marked in red correspond to the ones that are excluded by our method and those in green correspond to transitive edges that were not excluded. The simulated time delays are shown on the y-axis to the left of the heat map. z represents differences in time delays between inner paths and shortcut edges, with $z = 1$ corresponding to the most subtle difference possible for discrete data.

4.2.3 Dense networks and detection of shortcut pathways

In order to evaluate the ability of our model to detect transitive edges, we run a series of simulation experiments in which we include all transitive edges of the reference network. We set the time delays for the transitive edges to the expected time delay of corresponding inner pathways and subtracted the value z, where z is varying between 3 and 1. This yields a series of simulations with increasingly subtle differences in time delays between inner paths and shortcut edges, with $z = 1$ representing the most subtle differences possible for discrete data. Figure 4.9(A-D) shows the heatmaps of marginal posterior probabilities for average time delays. Again, most estimated average time delays are close to their target values, with a tendency to underestimate non-transitive edges and overestimate transitive edges. Moreover, the posterior distributions for transitive edges are more disperse. For low noise simulations ($\alpha = 0.0, 0.1$) and clear shortcuts (z= 2,3), the posterior distribution places hardly any weight on “x”. Hence all shortcut edges are clearly and correctly identified by our model. In high noise scenarios and for $z = 1$ posterior weight can accumulate in the edge exclusion state “x”. However, using a cutoff of 0.6 leads to only four missed shortcut edges in the whole set of simulation experiments. In the next section, we further investigate the stochasticity effect in a real application.

4.3 Application to cell differentiation in embryonic stem cells

We apply the DNEM approach to a data set on molecular mechanisms of self-renewal in murine embryonic stem cells. Ivanova *et al.*(2006) (17) used RNA interference techniques to downregulate six gene products associated with self-renewal regulatory function, namely *Nanog*, *Oct4*, *Sox2*, *Esrrb*, *Tbx3* and *Tcl1*. They combined perturbation of these gene products with time series of microarray gene expression measurements. Mouse embryonic stem cells (ESC) were grown in the presence of the leukemia inhibitory factor LIF thus retaining their undifferentiated self-renewing state (positive controls). Cell differentiation associated changes in gene expression were detected by inducing differentiation of stem cells through removing LIF and adding retinoic acid (RA) (negative controls). Finally, RNAi based silencing of the six regulatory genes was used in (LIF+, RA-) cell cultures to investigate, whether silencing of these genes partially activates cell differentiation mechanisms. Time series at 6-7 time points in one-day intervals were taken for the positive control culture (LIF+, RA-), the negative control culture (LIF-, RA+), and the six RNAi assays. In the DNEM framework the six regulatory gene products *Nanog*, *Oct4*, *Sox2*, *Esrrb*, *Tbx3* and *Tcl1* are *S-genes*, while all genes showing significant expression changes in response to LIF depletion are used as *E-genes*. Downstream effects of interest are those, where the expression of an *E-gene* is pushed from its level in self-renewing cells to its level in differentiated cells. Our goal is to model the temporal occurrence of these effects across all time series simultaneously.

4.3.1 Data preprocessing

We use log2 transformed values of MAS5.0 normalized data obtained from `www.nature.com/nature/journal/v442/n7102/suppinfo/nature04915.html`. In a comparison of the (LIF+, RA-) to the (LIF-, RA+) cell cultures 137 genes showed a greater than twofold up or down regulation across all time points. These were used as *E-genes* in our analysis. The two time series without RNAi were used to discretize the time series of perturbation experiments following a simple discretization method detailed in the next section, thereby setting an *E-gene* state to 1 in an RNAi experiment, if its expression value is far from the positive controls, and 0 otherwise. Genes that did not show any 1 after discretization across all experiments were removed, leaving 122 *E-genes* for further analysis.

4.3.2 Binary data

We transform the continuous expression data to binary values. We set an *E-gene* in a certain silencing experiment and time point to 1, if its expression value is sufficiently close to the negative controls, i.e. the intervention interrupted the information flow, otherwise we set it to 0. Let $C(i,k,s)$ denote the continuous expression measurement of E_k at time point t_s of a time series recorded after perturbation of S_i. Moreover, let $C^+(k,s)$ and $C^-(k,s)$ denote the corresponding measurements in positive and negative controls respectively. We set

$$D(i,k,s) = \begin{cases} 1 & \text{if} \quad C(i,k,s) < \kappa \cdot C^+(k,s) + (1-\kappa) \cdot C^-(k,s) \\ 0 & \text{otherwise} \end{cases} \tag{4.11}$$

κ can be optimized by varying its value from 0 to 1 and choosing the value where all negative controls are correctly recognized.

4.3.3 Time series analysis

We need binary data for each gene at each time point for each condition. Note that we have only three measurements per constellation: 1 negative control, 1 positive control and the measurement from the RNAi assay. In order to obtain robust estimates, data needs to be aggregated across time points. DNEMs assume that once a perturbation effect has reached an *E-gene*, it persists until the end of the time series. In other words, a one at time point t indicates that a downstream effect has reached the *E-gene* prior to t and not that it is still observable at this time. Hence, a typical discretized time series starts with zeros, eventually switches to ones and then stays one until the end of the series. We refer to these patterns as admissible patterns. For the vast majority of *E-genes*, the discretized data roughly follows admissible patterns. Nevertheless, exceptions are observed. We replace the time series for each gene by the closest admissible pattern, based on edit distances. In the case where several admissible patterns had the same edit distance to the time series, we chose the pattern holding the most ones. This curated data is used in further analysis.

4.3.4 Stability analysis

Since long computation times for Gibbs sampling prohibit the reconstruction of the network's topology from scratch using DNEMs, we used the triplet search approach for the standard nested effect approach (57) applied to the final time point to determine a topology for the network. Note, that the final time point of an admissible pattern accumulates information along the time series, because it reports a one whenever a downstream signal has reached the *E-gene* at any time. The binary data of the last time point across all *S-gene* perturbations is shown in Figure 4.10A, while Figure 4.10B shows the reconstructed network. A nested structure is visible. Our model is based on binary data, which requires gene expression profiles to be discretized. Discretization incurs a potential information loss. The inferred network structures can vary depending on the discretization threshold κ in equation (4.11) and so do the estimated average time delays. Nevertheless, in the application to the stem cell data described above, important network properties are stable: Most importantly, this applies to the central axis of the network

$$Nanog \rightarrow Sox2 \rightarrow Oct4 \rightarrow Tcl1,$$

and the domination of the network topology by feed-forward loops (transitive edges). In order to verify the robustness of these network features we run both the topology search using NEM and the time delay analysis using D-NEM on binary data produced with different settings of κ.

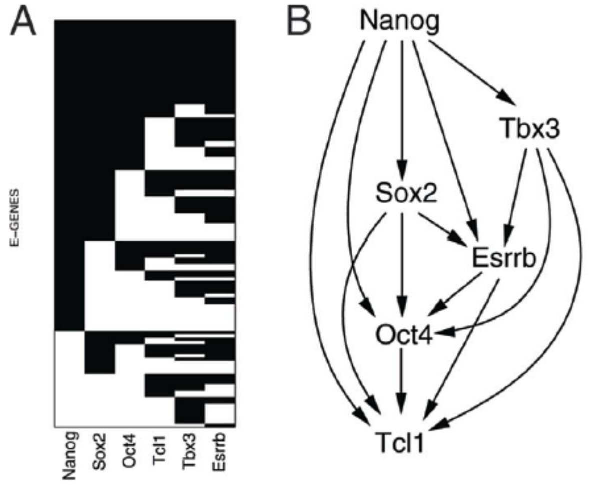


Figure 4.10: Stem cell data analysis - **A** Discretized data of the last time point across *E-genes* (rows) and *S-gene* perturbations (columns), with black representing downstream effects and white no effects. **B** The transitively closed nested effects model estimated from the data shown in **A** using static NEM.

4.3.5 Stability of network topologies in the static NEM analysis

For the topology search we vary κ from 0.4 to 0.9 in steps of 0.02 and count how often a certain edge was included into the estimated network. Figure 4.11(A) displays the relative frequencies of edges in a color coded adjacency matrix. White indicates 100% inclusion of the edge, black 0% inclusion, and gray indicate intermediate percentages. The areas framed in red highlight the stable structures of the network including the central axis from *Nanog* down to *Tcl1* and the hypothesis that both *Tbx3* and *Esrrb* act upstream of *Tcl1*. Network topologies further agree in that *Tbx3* and *Esrrb* are connected to the central axis. However, there is uncertainty with respect to the precise location of this cross talk edge as indicated by the gray tones in the *Tbx3* and *Esrrb* columns.

4.3.6 Stability of feed-forward loop detection in the DNEM analysis

We analyzed the effect of the discretization threshold κ on the DNEM analysis. Lower values of κ lead to more "1s" in the binary data and, hence, to smaller estimates of average time delays. The high number of feed-forward loops is a stable network feature across a wide range of thresholds. To demonstrate this, we run the DNEM algorithm on binary data with varying thresholds. Figure 4.11(B-E) shows the resulting posterior heatmaps for κ set to 0.6, 0.7, 0.8, and 0.9. Note that with the exception of the first simulation the analysis always excludes the same three edges from the network. It always yields a dense feed-forward loop dominated network. For thresholds of 0.6 and below (data not shown) the model becomes unstable.

The choice of κ is critical for the network analysis. Nevertheless, we observe network features that are remarkably robust with respect to the choice of κ. Notable are the central axis of the network from *Nanog via Sox2* and *Oct4* to *Tcl1*, and the domination of the network topology by feed-forward loops.

4.3.7 Decision between *Oct4* and *Tcl1* direction in network

The stable topology is based exclusively on the nesting of downstream effects. Time delays of signal propagation can now be used for fine tuning the topology: Originally, the NEM analysis suggested a bidirectional arrow between *Oct4* and *Tcl1* suggesting that the nesting of downstream effects in the final time point can not resolve the direction of interaction between these TFs. We fitted independent DNEM models for the two networks, which place *Oct4* up- or downstream of *Tcl1*. We used the deviance information criterion DIC (section 4.1.1.13) to decide which hypothesis is better supported by the observed time delays. The DIC strongly favors the model, which places *Oct4* upstream of *Tcl1*(DIC of 5491.1 compared to 5581.7).

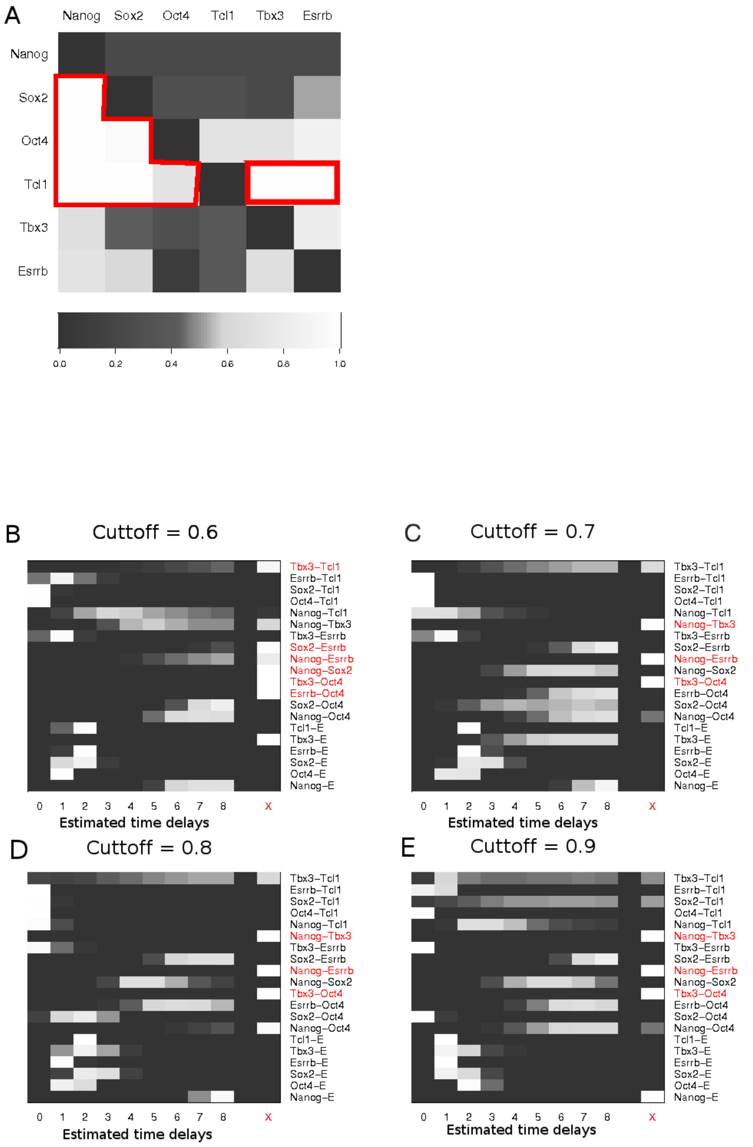


Figure 4.11: Stability Analysis - (A) *Topology:* Heat map of relative frequencies of edges when varying κ between 0.4 and 0.9 in steps of 0.02. White indicates 100% inclusion of the edge, black 0% inclusion, and gray indicates intermediate percentages. The areas framed in red highlight the stable structures of the network including the central axis from *Nanog* down to *Tcl1* and the hypothesis that both *Tbx3* and *Esrrb* act upstream of *Tcl1*. **(B-E)** *DNEM Analysis:* Heat map of the posterior distribution of average time delays for various cut-off discretized data (0.6-0.9). Rows correspond to edges of the network including those between S- and E-genes, while columns refer to average time delays. Edges in red represent excluded edges. Marginal posterior probabilities are gray-scale coded.

4.3.8 Convergence and Mixing analysis of the Gibbs sampler

Our analysis is based on a summary of the joint posterior distribution of all parameters as obtained from the Gibbs sampling trajectories. They are only valid if these samples represent the true posterior distributions. This is the case when the Gibbs sampler has converged to a stationary distribution and covers the whole posterior domain. In order to validate this, we test the convergence of the Gibbs sampler using three independent trajectories each starting from a random starting configuration. Median as well as 97.5% quantile Gelman and Rubin scale reduction factors are calculated for the first 1,000 iterations in windows of size 50. Figure 4.12 shows trace plots next to the corresponding convergence plots for 12 of the 20 rate constants. These are the 12 parameters for which we see non-deterministic posterior distributions. The trace plots show that the trajectories are swiftly moving through the full posterior domains. Moreover, in all 12 cases we observe fast convergence of the Gibbs sampler. After a burn in of at most 500 iterations the scale reduction factors stay within the interval $[1, 1.1]$. It is in the nature of models with discrete parameters that some parameters do not vary at all along the trajectories. This is the case for the 8 remaining rate constants not shown in Figure 4.12. In order to validate that this behavior of the Gibbs sampler is data driven and does not reflect trapping of the Gibbs sampler in a local configuration, we start 20 short Gibbs sampling trajectories of length 100 all with different random starting configurations. In all 20 trajectories we find the parameters converge to their stationary value after only 50 iterations. Moreover, they remain at this value for the rest of the trajectories. We notice, that the model parameters α and β might influence the observed convergence behavior. Setting one of them to a value below 0.01 compromises convergence. For higher values convergence is similar to that shown in the figure.

4.3.9 Inference of signaling in Network

Next, we exploit the DNEM Gibbs sampler trajectories associated with the network topology from Figure 4.10B to infer average time delays and regulatory control of *E-genes*. Figure 4.13A shows the histogram of average time delays (reciprocal rate constants) along the Gibbs sampling trajectory for the transitive edge between *Oct4* and its target *E-genes*. It is equivalent to the top most gray-scale intensity profile of the

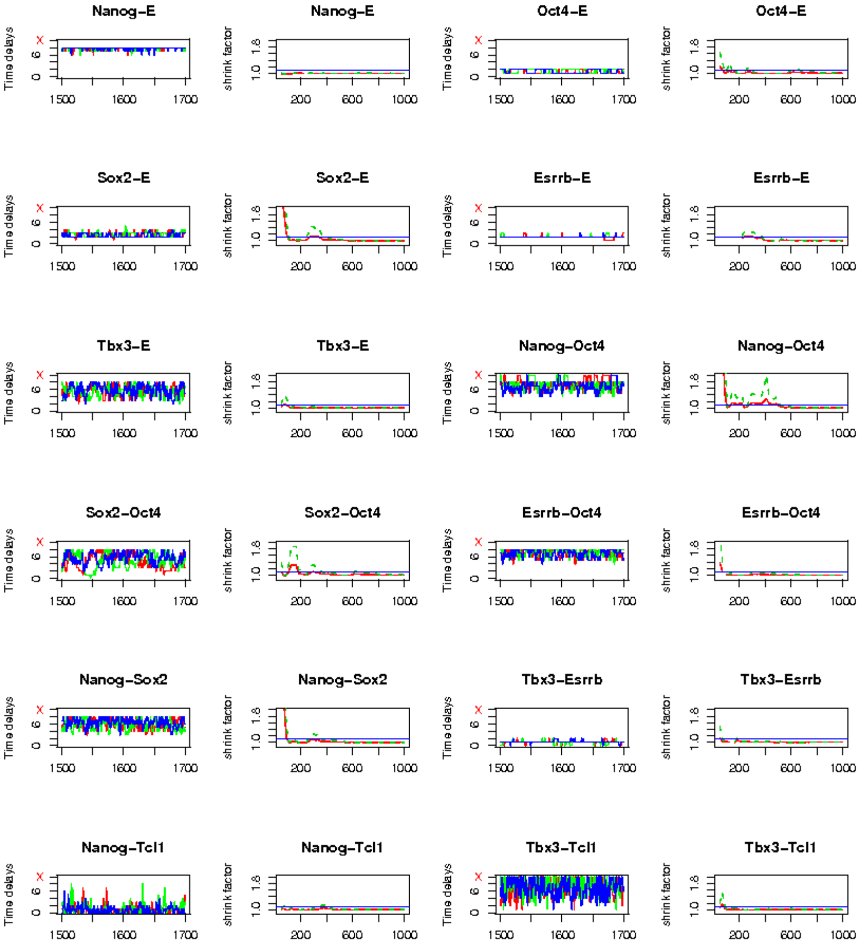


Figure 4.12: Diagnostic Plots for the Gibbs sampler - Shown are trace plots next to convergence plots for 12 of the 20 estimated average time delays (reciprocal rate constants). The trace plots hold 3 independent trajectories shown in different colors. The trajectories are swiftly moving through the full posterior domains. The convergence plots show median as well as 97.5% quantile Gelman-Rubin scale reduction factors calculated for the first 1,000 iterations in windows of size 50. After a burn in time of at most 500 iterations the scale reduction factors stay in the interval [1,1.1] marked by the blue horizontal line.

heat map in Figure 4.13B. The histogram reflects the marginal posterior probability of this parameter. The posterior heat map for all edges is shown in Figure 4.13B. Light gray indicates high marginal posterior probability while dark gray tones stand for low marginal posterior probabilities. The posterior mass either concentrates around zero indicating no time delay for this step of signal propagation, or intermediate values explaining secondary and tertiary effects, or high values with most of the posterior mass on κ_{T+1} (shown as x) suggesting that no signal is flowing through this edge. We exclude an edge if the posterior mass on κ_{T+1} is above 0.6. The resulting network is shown in Figure 4.13C. Strikingly, the time delay data provides evidence that all but three of the edges from Figure 4.10B actually transport signal. Note that the time delay data has also overruled the static NEM in one instance, in that it has removed the non-transitive edge between *Nanog* and *Tbx3*.

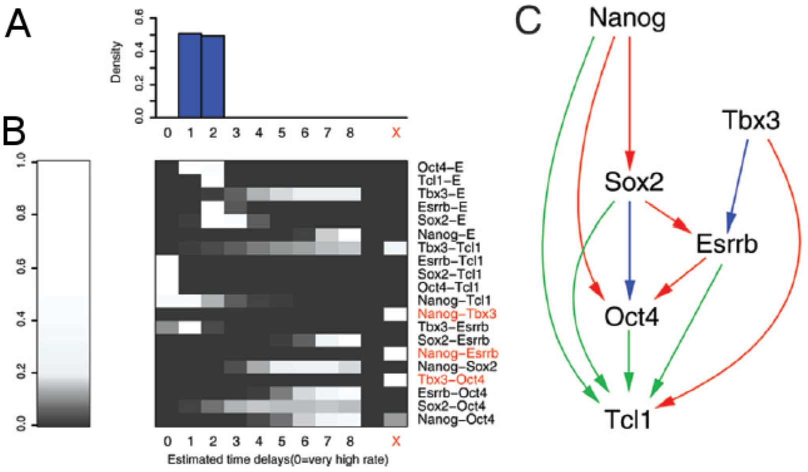


Figure 4.13: DNEM inference on signal propagation - A A histogram of the posterior probabilities for the average time delay associated with the edge from *Oct4* to its target *E-genes*. **B** Heat map of the posterior distribution of average time delays. Rows correspond to edges of the network including those between *S-* and *E-genes*, while columns refer to average time delays. Marginal posterior probabilities are gray-scale coded. The top row corresponds to the histogram described above. **C** The final network structure estimated by time delay analysis using DNEM. Edge colors correspond to estimated average time delays: fast signal propagation (green), intermediate signal propagation (blue) and slow signal propagation (red).

5

Cyclic Dynamic Nested Effects Models(CDNEMs)

Feedback circuits are important motifs in biological networks and part of virtually all regulation processes that are needed for a reliable functioning of the cell. This chapter extends DNEMs by allowing for the resolution of feedback loops in the signaling cascade. I demonstrate that cyclic DNEMs help reconstruct the unknown underlying network given time series data as well as infer the dynamics of the network. I first motivate the problem involved in the modeling of directed cyclic graphs in the context of DNEM and then use simulation studies to show the practical implementation of Cyclic DNEMs(CDNEMs). I further apply CDNEMs to data on molecular mechanism in early murine ESC development from Ivanova *et al.*(2006).

5.1 Model parameterization of CDNEMs

A cycle is a path with at least three edges, in which the first and last nodes are the same. Figure 5.1 gives an example of a directed cyclic graph with three nodes. In a directed cyclic graph, a set of edges which contains at least one edge (or arc) from each directed cycle is called a feedback arc set. Similarly, a set of vertices containing at least one vertex from each directed cycle is called a feedback vertex set. Edges $\{S_1S_2,S_2S_3,S_3S_1\}$ form a feedback arc set while $\{S_1,S_2,S_3\}$ is an example of a feedback vertex set. The cyclic DNEM problem can be formulated as shown in Figure 5.1 where we assume the S-genes to form feedback vertex sets with directed outdegrees of length

1 linking the E-genes. Assuming the same model parameterization as in DNEMs using (Θ, K), the goal is to generate the joint distribution of Θ and K using Gibbs sampling and then infer both the rates of signal propagation as well discriminate between direct and indirect signaling using the posterior samples.

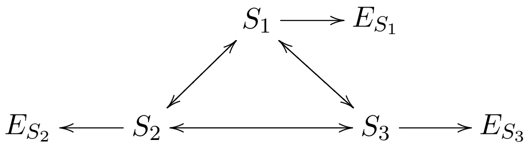


Figure 5.1: Cyclic DNEM model with 3 nodes - The cyclic DNEM model consists of a directed cyclic graph involving three S-gene nodes with feedback loops and three E-genes.

5.2 Probability density of signal propagation in a directed cyclic graph

Recall in DNEMs we are interested in the probability density function(pdf) of signaling times between two S-genes as well as between S-genes and E-genes. More precisely the pdf of signaling times between two nodes in a given directed acyclic graph (DAG). Extending DNEMs to handle cycles implies we need to enumerate all paths between an input node and an exit node in the cyclic graph to be able to estimate the pdf for signal propagation. Enumerating all paths between nodes in a cycle is not well defined without certain assumptions due to infinite looping. We establish boundary conditions such as, you can only visit each node once, or you cannot take the same path twice as in the “Travelling Salesman Problem” (106). The computational problem of enumerating paths in cycles have not been tackled in the framework of DNEMs.

5.2.1 Algorithm to enumerate all paths in a directed cyclic graph between two nodes

Assuming Figure 5.1 as the cyclic graph of interest we are interested to enumerate all paths from some S-gene S_i to some E-gene E_j. Note once the signal leaves an input S-gene S_i it can either activate its target E-gene E_i, or goes through other graph paths to other target E-genes linked to their S-genes respectively. Since there are cycles involved, and in general its impossible to enumerate all of them, we make use of atomic paths that don't loop and involve at most one cycle. I define an **atomic path** as a path that does not go through the same edge twice. Therefore an **atomic cycle** of node S_i is an atomic path that goes from node S_i and ends in node S_i. Atomic cycles only occur when enumerating paths from a certain S-gene S_i to its own E-gene E_i . In order to get all the atomic paths starting from node S_i to node E_j, we traverse the graph recursively from node S_i. While going through a child, we make a link child → parent in order to know all the edges we have already crossed. Before we go to that child, we traverse that linked list and make sure the specified edge has not been already walked through. When we arrive to the destination point, we store the paths we found. The following table gives a pseudocode algorithm for enumerating all atomic paths between two nodes in a directed cyclic graph. Note that looking for the atomic cycle of node S_i is the same as looking for the atomic path from S_i to S_i.

Table 5.1: Algorithm to enumerate all paths between two nodes in a graph - In order to get all the atomic paths starting from node A to node B, we traverse the graph recursively from node A.

Algorithm for enumerating all atomic paths between two nodes in a graph

```
procedure findallpaths(graph, start, end, path=()):
    path = path + start
    if start == end:
        return (path)
    if not graph.has.key(start):
        return ()
    paths = ()
    for node in graph(start):
        if node not in path:
            newpaths = findallpaths(graph, node, end, path)
            for newpath in newpaths:
                paths.append(newpath)
    return paths
```

5.2.2 Probability density of signal propagation in CDNEMs

In the last chapter, we estimated the joint posterior of rate constants in a directed acyclic graph Φ assuming independent exponential time delays with varying signal propagation rates on the edges. The distribution for a fixed linear path in a given network is given by Equation 4.1. In the case of directed acyclic graphs with several alternating paths between nodes the cdf can be approximated by Equation 4.2. We now consider the case of directed cyclic graphs. Given a directed cyclic graph like the one shown in Figure 5.1, we would like to estimate the cdf from a certain input node say S_i to some output node E_{S_j}. Signal propagation can be from S_i to its target E-gene E_{S_i} or to another E-gene E_{S_j}. The paths between these two nodes consist of atomic paths and cycles. I illustrate with example that Equation 4.1 can still be used to estimate the cdf of signaling times between nodes S_i and E_{S_j} even when atomic cycles are involved. Lets consider the nodes S_1 and E_{S_1} from Figure 5.1. The atomic paths between S_1 and E_{S_1} are $\{(S_1 - S_2 - S_3 - S_1 - E_{S_1}), (S_1 - S_3 - S_2 - S_1 - E_{S_1}), (S_1 - E_{S_1})\}$. Note that edge $(S_1 - E_{S_1})$ occurs in all three paths and is associated with a certain time delay. If we assume exponential time delays for all edges in the graph, the expectation of the distribution for signal propagation between S_1 and E_{S_1} will be equal to the expectation of signaling time corresponding to the edge $(S_1 - E_{S_1})$ only. This is because we add something positive to all the time delays associated with the two alternating atomic cycles $\{(S_1 - S_2 - S_3 - S_1), (S_1 - S_3 - S_2 - S_1)\}$. In general, if we have n independent positive random variables $\{X_i, i = 1, ..., n\}$, and X_j also a positive random variable with $i \neq j$ then,

$$E(min(X_j, X_1 + X_j, ..., X_n + X_j)) = E(X_j).$$

$E(X_j)$ provides an expected value for the random variable which corresponds to the minimum of two or more random variables. In practice, this implies we do not need to enumerate all the paths from S_1 to E_{S_1} to calculate the cdf of signaling time in this path. We only approximate the distribution for signal propagation between S_1 and E_{S_1} to be exponential with a certain rate constant. This makes sense since we expect the first blocked signal from S_1 to activate E_{S_1} fastest compared to signals from alternating paths. This approach to estimate the cdf for signal propagation time between S_1 and E_{S_1} can be generalized to the cdf of signaling times corresponding to paths between S_i and E_{S_j}, $i \neq j$ consisting of atomic cycles. From Figure 5.1 notice that the shortest path between S_i and E_{S_j} is still linear with corresponding cdf of signaling time estimated by Equation 4.1. Hence, we expect the time delays for the atomic paths between S_i and E_{S_j} to be faster than any alternating path with cycles.

5.3 Simulation results

We demonstrate the performance of cyclic DNEM in various simulation scenarios with artificially generated data based on different model assumptions. We show that the cyclic DNEM can be used to make inferences on both the underlying biological network as well the dynamics of signal flow in the unknown network.

5.3.1 Data generation

We evaluate our method in the context of simulated data from the cyclic network shown in Figure 5.1. We parameterize the graph as shown in Figure 5.2. Note that the network is transitively closed. Time delays for signal propagation between *S*- and *E-genes* are set to 1. For signal propagation between $S - genes$ we simulate 4 different scenarios summarized in Table 5.2 corresponding to Figures 5.3(A-D). The rows represent the simulated time delays corresponding to the edges. Column 2 represents the situation with all time lags set to 1 unit time. Here we have subtle differences in time delay between transitive edges and non-transitive edges. Column 3 corresponds to the situation with much smaller time delays for the transitive edges compare to their non-transitive counter parts. Column 4 represents the situation when the underlying network is directed acyclic and column 5 corresponds to the situation when the pathway is a single cycle in one direction. For all *E-gene* positions the expected data pattern across time points and perturbation experiments is calculated and artificial *E-gene* data is simulated by adding independent binary noise to these patterns using a range of different noise levels: $\alpha = 0.0, 0.1, 0.2, 0.3$ and $\beta = 1/2\,\alpha$. We simulate data for 20 *E-genes* per *S-gene* and one measurement per time point, resulting in a data array of 960 binary values. CDNEM is run on this data using two independent runs of 5,000 iterations, from which the first 2500 are discarded as burn in time, leaving 5,000 posterior samples per simulation. For calculating the likelihood, we set the tuning parameters as in the DNEM scenario with α and β set to 0.2 and 0.1 respectively.

5.3.2 Results

Figure 5.4 shows the heatmaps of marginal posterior probabilities for average time delays for various network scenarios with $\alpha = 0.0, 0.1, 0.2, 0.3$. Light gray indicates high marginal posterior probability while dark gray tones stand for low marginal posterior

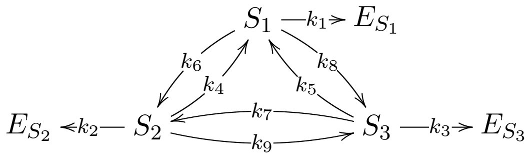


Figure 5.2: Parameterization of the Cyclic DNEM model with 3 nodes - The cyclic DNEM model consists of a directed cyclic graph involving three S-gene nodes with feedback loops and three E-genes. The 9 rates on the edges form the model parameters.

Table 5.2: Simulated time delays for edges in cyclic graph - The rows represent the simulated time delays. Column 2 represents the situation with 1 unit time lag for all edges. Here we have subtle differences in time delay between transitive edges and non-transitive edges. Column 3 corresponds to the situation with larger time delay differences between existing transitive edges compared to their non-transitive counter parts. Column 4 represents the situation when the underlying graph is DAG and column 5 represents the situation when the simulated pathway is directed and forms a cycle.

Rates	Equal delays	Dense network	Directed acyclic	Cycle
k_1	1	1	1	1
k_2	1	1	1	1
k_3	1	1	1	1
k_4	1	2	100	1
k_5	1	3	100	100
k_6	1	3	1	100
k_7	1	3	1	1
k_8	1	2	1	1
k_9	1	1	100	100

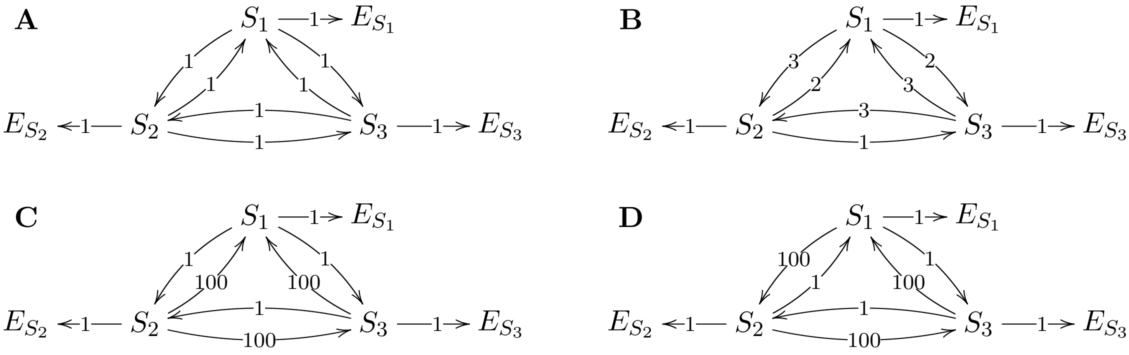


Figure 5.3: Directed cyclic graphs with simulated time delays - Simulated time delays for edges in the cyclic graph Figure 5.2 under various scenarios. **A** represents the situation with all time lags set to 1 unit time. **B** corresponds to the situation with much smaller time delays for the transitive edges compare to their non-transitive counter parts. **C** represents the situation when the underlying graph is DAG. We set the time delay for non-existent edges to 100. **D** corresponds to the situation when the simulated pathway is a single cycle in the direction of $S_1S_3S_2S_1$.

probabilities. Most estimated average time delays are close to their target values with more variability occurring in the scenarios involving longer simulated time delays on the edges. In general most edges are clearly and correctly identified by our model. We go through the specific properties of each simulation study.

5.3.2.1 Fully connected network with equal time delays

In order to evaluate the ability of our model to detect distinct edges even when signal flow is bi-directional and the time delays between signaling nodes are equal, we run a simulation with all rate constants set to 1. This represent subtle differences in time delays between direct and indirect signals. Figure 5.4A summarizes the posterior distribution for all rate parameters. All edges are clearly and correctly identified by our model.

5.3.2.2 Cyclic dense network with varying time delays

We next vary the expected time delays between transitive and non-transitive edges using values corresponding to column 3 from Table 5.2. In other words we simulate data from a dense cyclic network with several FFLs and feedback graphs(FBLs). For

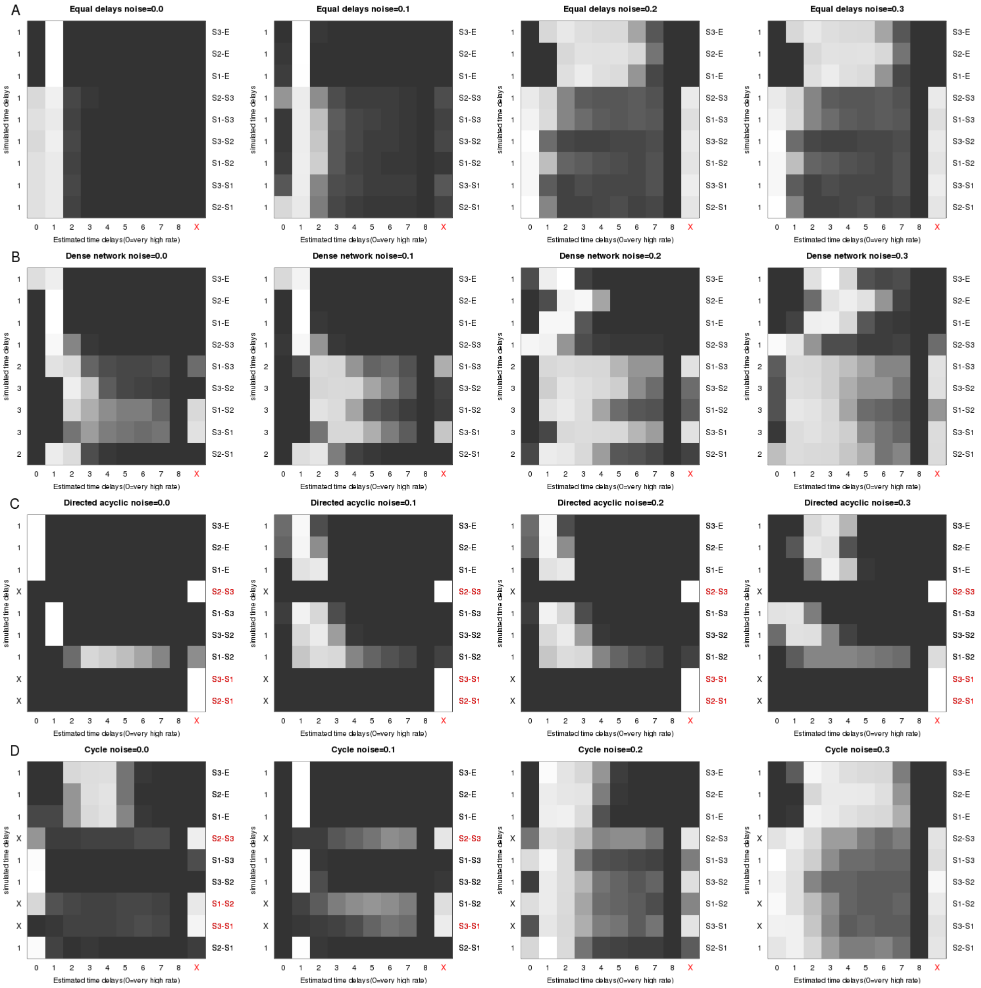


Figure 5.4: Heat map of the posterior distribution of CDNEM model - (A-D) The heatmap correspond to simulated scenarios corresponding to a network with equal time delays, a dense network, directed acyclic network, and a cyclic network. Rows correspond to edges of the network including those between S- and E-genes, while columns refer to average time delays. Marginal posterior probabilities are gray-scale colored with light gray indicating high and dark gray indicating low probability. Edges marked in red correspond to the ones that are excluded by our method. The simulated time delays are shown on the y-axis to the left of the heat map.

low noise simulations ($\alpha = 0.0, 0.1$), the posterior distribution places hardly any weight on “x” keeping all edges in the model Figure 5.4B. In a nutshell the model was able to retain all edges even in the presence of high noise.

5.3.2.3 Directed acyclic networks

The next scenario was to investigate the ability of our model to detect the presence of a directed acyclic pathway even when the input graph is cyclic. We set the expected time delays for k_4, k_5, and k_9 to 100 making the corresponding edges practically nonexistent. To investigate this scenario using a cutoff of 0.6 for the “x” state leads to the three excluded edges as desired even for high noise levels Figure 5.4C. There is a tendency to underestimate rate parameters for perfect data. In general, we are able to detect the underlying direct acyclic graph in the presence of noisy data.

5.3.2.4 Network with only one cycle

Finally we examine the situation where the underlying graph is directed and has one cycle $S_1 - S_3 - S_2 - S_1$. We set k_5, k_6, and k_9 to 100 in this case making the reverse cycle $S_1 - S_2 - S_3 - S_1$ practically non-existent. At low noise levels, Figure 5.4D shows that our model puts a high weight on the “x” state for these edges thereby kicking them out of the model. The remaining edges form a directed cycle as desired.

5.4 Application of CDNEMs to cell differentiation in embryonic stem cells

We apply the cyclic DNEM approach to the same preprocessed binary data set on molecular mechanisms of self-renewal in murine embryonic stem cells from Ivanova *et al.*(2006) (17) used in the last chapter. We demonstrated using Figure 4.11 a nested structure of effects at the last time point between *Nanog*, *Oct4* ,*Sox2*, and *Tcl1* which form a linear cascade which puts *Nanog* on top forming a cascade from *Nanog* to *Sox2* to *Oct4* and finally to Tcl1. We also showed that both *Tbx3* and *Esrrb* act upstream of *Tcl1*. However its not clear how *Tbx3* and *Esrrb* fit in the cascade and whether there exist FFls or Feedback loops as well. Instead of using only a directed acyclic graph we used a cyclic network with bidirectional edges between all nodes as input graph Figure 5.5. In other words we used a closed network assuming all edges present. In all we have 36 rate parameters to update. All other model parameter settings for the Gibbs sampler are kept the same as in the DNEM scenario. To speed up convergence we used initial parameter values from the CDNEM with the non-stochastic constant time delays approach. Figures 5.6 and 5.7 show the posterior heat map for all 36 edges under nonstochastic and stochastic signaling assumptions respectively.

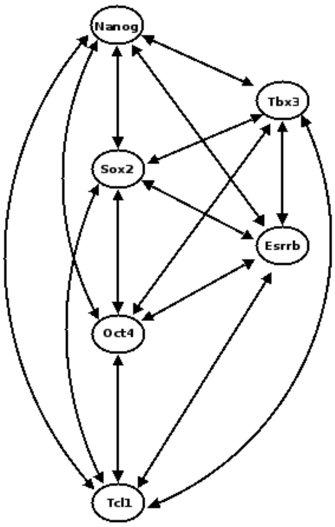


Figure 5.5: Fully connected directed input graph with 6 key TFs - A fully connected cyclic network with bidirectional edges between all regulatory proteins as input graph for CDNEM. There are 36 rate parameters to update corresponding to the number of edges.

5.4.1 Inference of signaling in Network during early ESC differentiation

We exploit the CDNEM Gibbs sampler trajectories associated with the fully connected network topology for all possible edges to infer average time delays and regulatory control of *E-genes*. The heatmap in Figure 5.6 summarizes the posterior distribution for all parameters for the non-stochastic scenario while Figure 5.7 gives the complete picture under stochastic modeling of time delays. Recall light gray indicates high marginal posterior probability while dark gray tones stand for low marginal posterior probabilities. There is an apparent higher evidence for mixing and dispersion of the posterior distribution under the model with stochastic assumptions. On the contrary expected posterior modes are more conspicuous in the non-stochastic case as expected. Similar

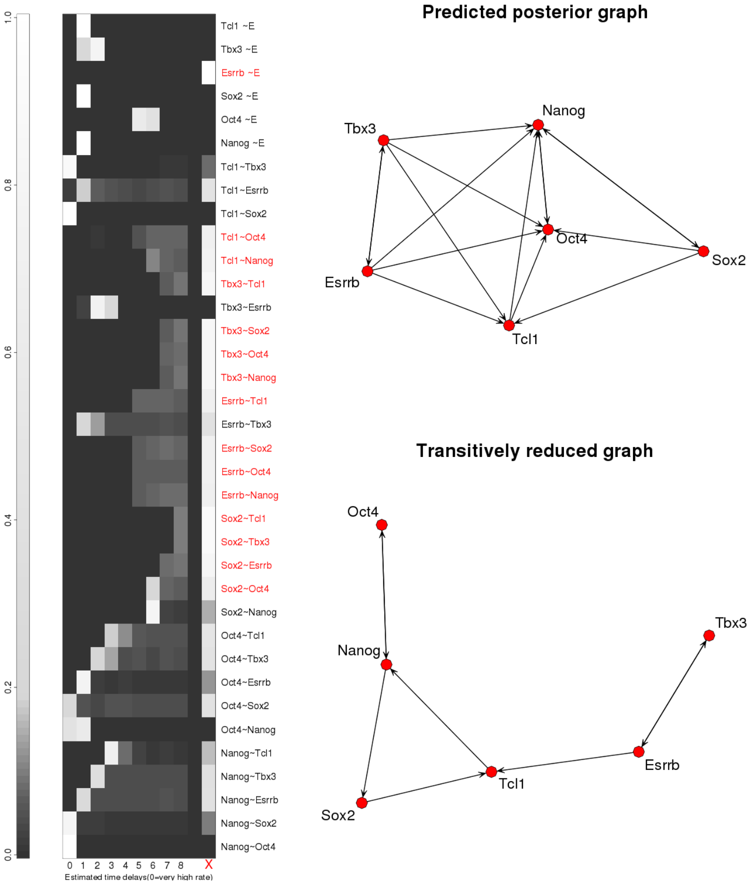


Figure 5.6: Heatmap of posterior distribution and output graph of signal flow under non-stochastic assumptions - Heat map of the posterior distribution of average time delays under non-stochastic signaling. Rows correspond to edges of the network including those between *S*- and *E-genes*, while columns refer to average time delays. Marginal posterior probabilities are gray-scale coded. The graph on the top right corresponds to the predicted graph that supports the data best. A transitively reduced version of the graph can be seen at the bottom right

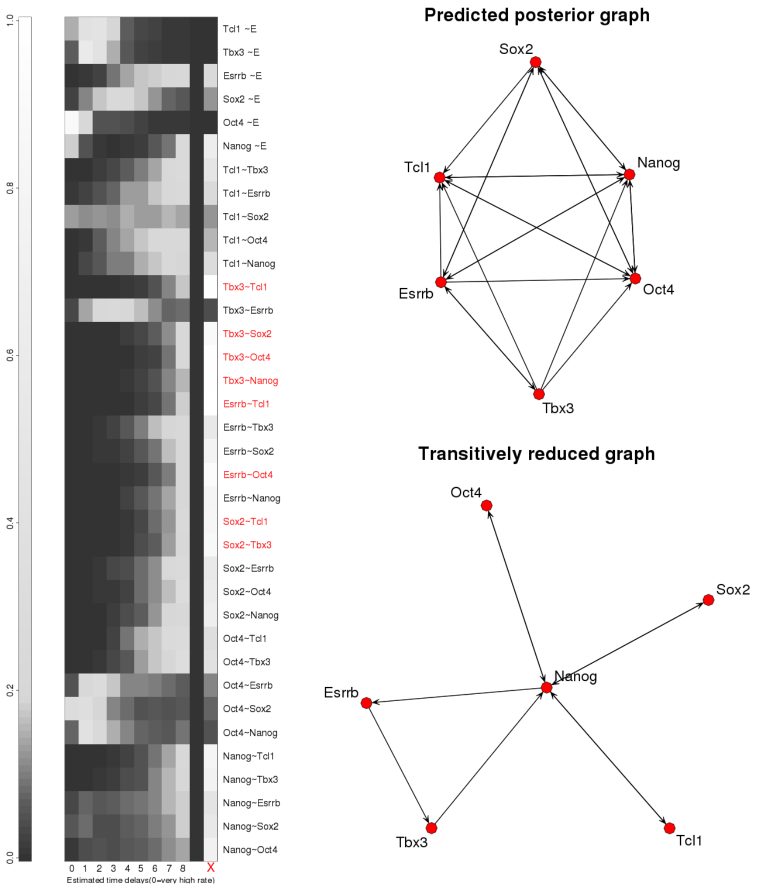


Figure 5.7: Heatmap of posterior distribution with output graph of signal flow under stochastic signaling - Heat map of the posterior distribution of average time delays under stochastic modeling. Rows correspond to edges of the network including those between *S*- and *E-genes*, while columns refer to average time delays. Marginal posterior probabilities are gray-scale coded. The edges in red are non-existent. The graph on the right corresponds to the predicted graph that supports the data best. Transitively closing this graph gives the topology at the bottom right.

to the results from the DNEM model, the posterior mass either concentrates around zero indicating no time delay for this step of signal propagation, or intermediate values explaining secondary and tertiary effects, or high values with most of the posterior mass on κ_{T+1} (shown as x) suggesting that no signal is flowing through this edge. Using a cut-off of 0.6 for x, there are a few interesting observations from the joint posterior distribution from both scenarios. Firstly, only a few number of edges have been kicked out of the model especially in the stochastic situation thereby confirming a very dense network with both FFLs and FBLs involved in early ESC development. In addition the edge $Nanog \rightarrow Tbx3$ has been kicked out of the model like in the DNEM situation. Also there is a high evidence of signal communication about expected time delay of 2 days. This goes to support the fact that early stage differentiation occurs after about 2 days (18). Furthermore, we see that most of the edges that are kicked out correspond to those edges associated with *Tbx3* and only one of the kicked out edges involves *Nanog*. Thus *Nanog* is acting as a key sensitizer for stem cell differentiation. The resulting network under stochastic signal transduction is the graph on the top right in Figure 5.7. A transitively reduced network at the bottom right of Figure 5.7 shows that Nanog is highly connected to all the other regulators. In a nutshell, the time delay data provides evidence that all but a few number of the edges from the input network actually transport signal. Note that the time delay data has also overruled the static NEM and DNEM models in several instances especially the signal flow between *Nanog*, *Sox2* and *Oct4* in one direction. It seems signal flow between these core TFs involves both FFLs and FBLs.

6

Impact of Dynamic Nested Effects Models

My work has already been taken up and extended by others especially in the direction of improving the running time of DNEMs. In the following, I summarize the paper of Frölich *et al.*(2010) (107) pointing out the cross links and conceptual differences to my own work.

6.1 Fast Cyclic Dynamic Nested Effects Models(FCDNEMs)

Due to the long running times needed in Gibbs sampling for DNEMs it will not be feasible in practice to infer dynamics of very large networks. Frölich *et al.*(2010) introduce a parallel approach of CDNEMs which circumvents the time consuming Gibbs sampling step for inference of signal propagation rates on the edges of a network(107). This approach does not aim to infer the rates of signaling. It only estimates the time lag between a perturbation and an observed downstream effect, there by providing the possibility to unroll the signal flow in the upstream signaling cascade over time. It uses a simple greedy hill climbing strategy (section 2.3.3) in combination with a non-parametric bootstrap to assess confidences of inferred edges. The formulation of this dynamic model is just an extension of NEMs to handle cycles. Cycles in Φ imply that perturbation effects are indistinguishable within this model. However, we already showed that time series measurements of perturbation effects help resolve biological feedback loops and distinguish between direct and indirect effects.

6.1.1 Model parameters for FCDNEMs

Similar to the original DNEMs in chapter 4, let $D(i, k, l, s)$ denote the expression measurement of E_k in time point t_s of the l'th replication of a time series recorded after perturbation of S_i. t_s is replaced with t corresponding to the *index* of time point in a discrete time series, not the time point itself. These measurements could be p-values, counts or any other kind of statistics quantifying the effect of a knock-down for E-gene E_k under perturbation of S-gene S_i at time t. Suppose the true underlying pathway is given by Figure 6.1A. The signal flow is unrolled in this network over time (Figure 6.1B) in the following way: The node set $\mathcal{E}(t) = \{ E(t), E \in \mathcal{E}\}$, $\mathcal{S}(t) = \{ S(t), S \in \mathcal{S}\}$ of the dynamic network consists of a copy of the static network nodes, one for each time point $t = 1, ..., T$. An E-gene $E(t)$ is linked to $S(t)$ whenever E is linked to S in the static situation, i.e, it is determined by the same matrix $\Theta = |\mathcal{S}| \times |\mathcal{E}|$ as in the static case following (59). The actual unrolling takes place in the wiring of the S-genes. Informally, the static adjacency matrix Φ is converted to a $|\mathcal{S}| \times |\mathcal{S}|$ weighted adjacency matrix Ψ $= (\psi_{ij})$, where 0 means no edge and a value $\psi_{ij} > 0$ implies an influence of node i on E-genes downstream of node j delayed by ψ_{ij} time steps. Specifically, $T \geq \psi_{ij} \geq \Phi_{ij}$ for $i, j \in \mathcal{S}$. A non-zero entry ψ_{ij} implies that there are edges $S_i(t) \rightarrow S_j(t + \psi_{ij})$, t $= 1, ..., T - \psi_{ij}$. Furthermore, the convention $\psi_{ii} = 1$ is made. A positive time lag between nodes i and j in the model describes the number of time steps, after which a knock-down of node i results in an observed effect downstream of node j. This implies there are no assumptions made about the physical time it takes a signal at node j to produce a downstream effect at an E-gene. In contrast to classical Dynamic Bayesian Networks (108), an edge in the model may not connect consecutive time layers, but it may skip a certain amount of time steps (as it is the case for the entry $\psi_{S_2 S_3} = 2$ in Figure 6.1B, which implies the edge $S_2(1) \rightarrow S_3(3)$. In other words, the model does not rely on a first order Markov assumption. In this way the unknown and variable time delays in perturbation responses are modelled due to the upstream signaling. In the following I refer to the model as FCDNEM.

6.1.2 Marginal likelihood for discrete model

Considering the same parameterization like in static NEMs given in chapter 2, and assuming independence of time point measurements, the marginal likelihood Equation 2.6

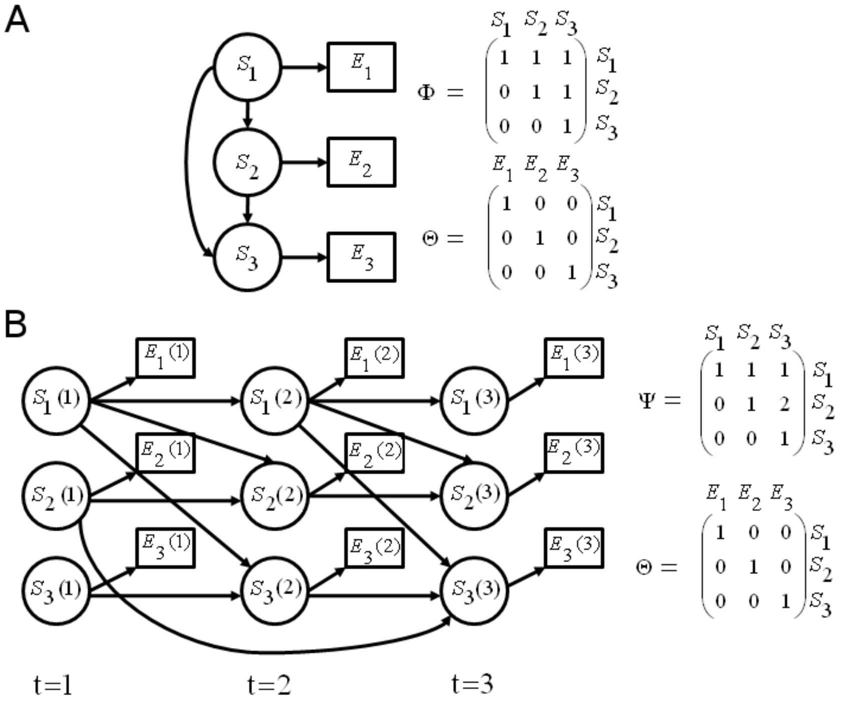

A
S_1
S_2
S_3
E_1
E_2
E_3
$\Phi = \begin{array}{cccc} S_1 & S_2 & S_3 & \\ 1 & 1 & 1 & S_1 \\ 0 & 1 & 1 & S_2 \\ 0 & 0 & 1 & S_3 \end{array}$
$\Theta = \begin{array}{cccc} E_1 & E_2 & E_3 & \\ 1 & 0 & 0 & S_1 \\ 0 & 1 & 0 & S_2 \\ 0 & 0 & 1 & S_3 \end{array}$
B
$S_1(1)$
$S_2(1)$
$S_3(1)$
$E_1(1)$
$E_2(1)$
$E_3(1)$
$S_1(2)$
$S_2(2)$
$S_3(2)$
$E_1(2)$
$E_2(2)$
$E_3(2)$
$S_1(3)$
$S_2(3)$
$S_3(3)$
$E_1(3)$
$E_2(3)$
$E_3(3)$
$\Psi = \begin{array}{cccc} S_1 & S_2 & S_3 & \\ 1 & 1 & 1 & S_1 \\ 0 & 1 & 2 & S_2 \\ 0 & 0 & 1 & S_3 \end{array}$
$\Theta = \begin{array}{cccc} E_1 & E_2 & E_3 & \\ 1 & 0 & 0 & S_1 \\ 0 & 1 & 0 & S_2 \\ 0 & 0 & 1 & S_3 \end{array}$
t=1
t=2
t=3

Figure 6.1: Standard NEM with 3 nodes - A static NEM is parameterized by a directed graph between S-genes encoded by Φ, together with a directed graph attaching each E-gene to an S-gene given by Θ. **B** Unrolling of the signal flow in the network from **A** along time. This corresponds to the network topology and parameterization of FCDNEM.

is extended to include time as :

$$p(D|\Psi, \Theta) = \prod_{i \in \mathcal{E}} \sum_{s \in \mathcal{S}} \prod_{l \in \mathcal{L}} \prod_{t=1}^{T} p(D_{il}(t)|\Psi, \Theta_{is} = 1) P(\Theta_{is} = 1) \tag{6.1}$$

To compute $p(D_{il}(t)|\Psi, \Theta_{is} = 1)$ according to the proposed unrolling of the signal flow, a time dependent Boolean perturbation state for each S-gene s is introduced, which encodes an active state when perturbed as 0 and 1 when unperturbed. A knock-down of s corresponds to a switch $1 \rightarrow 0$. Since the perturbation state of s at a particular time step t is not observable, we identify it with the value $[s(t)]$ of a random variable $s(t)$. Let $pa(s)(t)$ denote the set of parents nodes of s at time t (i.e. the set $\{p|0 < \psi_{ps} < t\}$; which can be empty, if appropriate). Then, according to the unrolling of the signal flow over time, we write:

$$\begin{aligned} p(D_{il}(t)|\Psi, \Theta_{is} = 1) &= \sum_{[s(t)] \in 0,1} p(D_{ikl}(t)|s(t) = [s(t)], \Theta_{is} = 1) \\ &\times \; P(s(t) = [s(t)]|pa(s)(t)) \end{aligned} \tag{6.2}$$

In the absence of more precise information we define:

$$\begin{aligned} P(s(t) = 0|pa(s)(t) = [r]) &= \begin{cases} 1 & \exists p \in pa(s)(t) : [p] = 1 \\ 0 & \text{otherwise} \end{cases} \\ P(s(t) = 1|pa(s)(t) = [r]) &= 1 - P(s(t) = 0|pa(s)(t) = [r]) \end{aligned} \tag{6.3}$$

The above definition can be interpreted as s is perturbed at time t, if any of its parents (including s itself) are perturbed. Assuming independence of observations the marginal likelihood $p(D_{ikl}(t)|s(t) = [s(t)], \Theta_{is} = 1)$ can be calculated using the methods of static NEMs discussed in chapter 2.

6.1.3 Using Priors for network structures and time delays

In the last chapter a weighted adjacency matrix Ψ is introduced as a summary representation of a given network structure and time delays between S-genes and E-genes. Learning the structure of Φ is equivalent to learning the matrix Ψ based on the likelihood given in Equation 6.2. While scoring a given network, we assume observing an effect after longer time delays is less likely smaller time delays. Moreover redundant edges are left out of the model since they do not change the likelihood of the model.

These considerations are taken into account during the specification of $P(\Psi)$. Following Floerich *et al.*(2007) (58), prior probabilities for each edge are specified as follows :

$$p(\Psi|\nu) = \prod_{i,j} \frac{1}{2\nu} \exp \frac{-|\psi_{ij} - \hat{\psi}_{ij}|}{\nu}$$

where $\nu > 0$ is an adjustable scaling parameter. The parameter ν can be chosen according to the **BIC** criterion(109):

$$BIC = -2 \log p(D|\Phi) + \log(|\mathcal{E}|) \sum_{i,j} \mathbf{1}|\psi_{ij} - \hat{\psi}_{ij}| > 0$$

where $\sum_{i,j} \mathbf{1}|\psi_{ij} - \hat{\psi_{ij}}| > 0$ is an estimate of the number of parameters in the model. Usually we favor sparse network structures.

6.1.4 Network Learning for FCDNEMs

Learning the network structure Φ that fits the data best is equivalent to finding an optimal weighted adjacency matrix Ψ where the entries of Ψ_{ij} can take discrete values 0, ..., T . The greedy hill climbing strategy(section 2.3.3) is used . By this approach three search operators are used: edge weight increase ($\Psi_{ij} \mapsto \Psi_{ij} + 1$, if $\Psi_{ij} < T$), edge weight decrease ($\Psi_{ij} \mapsto \Psi_{ij} - 1$, if $\Psi_{ij} > 0$), edge reversal (exchange of Ψ_{ij} and Ψ_{ji}). At each step we apply all possible operators and accept the solution that increases the posterior likelihood most. This requires $O(|S|^2)$ likelihood evaluations per search step, where each likelihood computation according to Equation 6.2 has a time complexity of $O(T|\mathcal{E}||S|^2)$ on its own. Hence each search step requires $O(T|\mathcal{E}||S|^4)$ time. This is much faster than using the Gibbs sampling approach.

To further assess the confidence of the inferred network hypothesis on real experimental data, non-parametric bootstrapping (1000 times) is used. Thus, from the whole set $\mathcal{E}$ of available downstream effects bootstrap samples $\mathcal{E}' \subset \mathcal{E}$ of size $|\mathcal{E}|$ are randomly drawn with replacement. On each bootstrap sample a network hypothesis using greedy hill climbing is estimated. This allows the estimation of confidence intervals for each Ψ_{ij}.

6.2 Application of FCDNEMs to cell differentiation in embryonic stem cells

FCDNEMs is applied to our famous preprocessed dataset by Ivanova *et al.*(2006)(17) within a non-parametric bootstrap procedure and how often each edge appears in 1000 inferred networks (one network per bootstrap sample) is recorded. The exact binomial distribution 95% confidence intervals is computed for the appearance probability of each edge via R-package binom (110). Only edges with lower confidence bound > 50% are regarded as reliable and shown in Figure 6.2. The median time lags for all edges is 1. There are several similarities to the inferred network shown in Figure 4.13, which was obtained via the DNEM method, namely the cascades $Tbx3 \rightarrow Esrrb \rightarrow Oct4 \rightarrow Tcl1$, $Nanog \rightarrow Oct4 \rightarrow Tcl1$ and $Sox2 \rightarrow Oct4 \rightarrow Tcl1$. A further striking similarity is that the transcription factor $Oct4$ regulating $Tcl1$ is itself jointly regulated by the three transcription factors $Nanong$, $Sox24$ and $Esrrb$. In contrast to model from DNEM, $Nanog$ is not placed upstream of $Sox2$ and does not have any indirect outgoing edges. Indeed, the only shortcut in this network is $Sox2 \rightarrow Tcl1$. This network is thus very much sparse than the ones from Figure 4.13 and Figure 5.7. This is probably due to the strong influence of the network prior. However all the predicted edges occur in the CDNEM predicted network (Figure 5.7) as well and there are no feedback loops. Comparing results from Figures 4.13, 5.7 and 6.2 we see that Figure 5.7 gives a complete picture of molecular signaling in early ESC development involving both FFLs and FBLs.

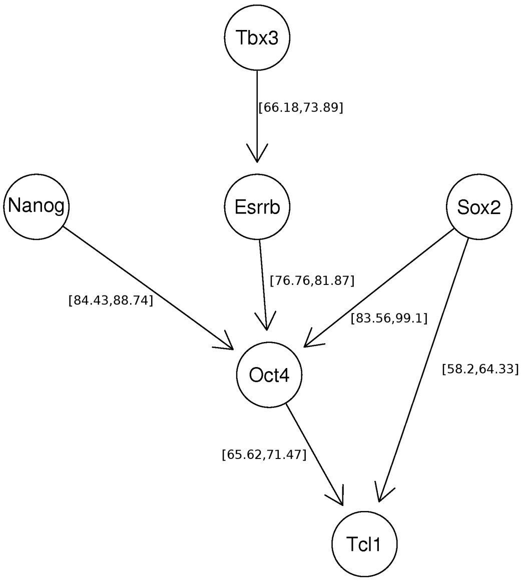


Figure 6.2: Inferred network for murine stem cell development - Inferred network for murine stem cell development with 95% confidence intervals for the presence of the edges.

7

Summary and Outlook

Time series RNA interference (RNAi) is an effective tool for genome-scale, high throughput analysis of genes, that are important for specific phenotypic traits of interest. The temporal and spatial placement of these genes in signal transduction pathways or developmental transcriptional networks remain a challenge as well as understanding the dynamics of signal flow in the given network. Since direct observations of intervention effects on other pathway components are often not available, large-scale datasets such as RNAi screens may only contain information of secondary or tertiary downstream effects. This dissertation develops methodology to show that by observing the nested structure of significant up or down regulations of affected genes over time, we may reverse engineer features of the upstream signaling pathway. It tackles two important problems involved in time series perturbation data.

1. Given a biological pathway topology and time series silencing data, how do we infer the signaling dynamics between pathway components from the data
2. Given only the time series perturbation data how can we make inferences on both the underlying biological network as well the dynamics of signal flow in the unknown network.

7.1 Conclusions

I introduced a new methodology called Dynamic Nested Effects models(DNEMs) which is an extension of NEMs to handle time series perturbation data. DNEMs allow the dissection of biological processes into signaling and expression events, and the analysis of cellular signal flow. In an application to decision making in mural embryonic stem cell development, I could show that a feedforward loop dominated gene regulation network ensures that cell differentiation is a quasi unidirectional process in vivo. However this model assumes that the underlying network is directed acyclic which is a limitation since feedback loops are essential motifs in developmental regulatory networks. I extended the methodology of DNEMs to cyclic DNEMs(CDNEMs) showing that even when the underlying network is unknown, CDNEMs can both reconstruct the unknown network as well as decode the dynamics involved in the network. I was able to unravel such a molecular communication in embryonic stem cells of the mouse.

The results from this thesis contribute to our understanding how stem cells succeed to carry out differentiation to specialized cells of the body such as muscle cells or neurons of the brain, a process that goes more or less only in one direction. The signaling processes are connected together such that a negligible reduction of the concentration of a key molecule named NANOG releases a signal, that is reinforced in the network, thereby initiating the differentiation of cells. Simultaneously the organization of other key players in the entire differentiation process makes the reverse process no longer possible even with slight increase in NANOG concentration by chance. The FFLs in the network stabilize the differentiated state of cells relative to self renewal by filtering out random fluctuations. The feedback loops implement memory of an input signal, even after the input signal is gone. A reversal of the differentiation process would cause a latent cancer risk. This reconstructed network of molecular communication proposes how organisms protect themselves against the reversal of cell differentiation and thereby against cancer.

In general, DNEMs can be used to model the dynamics of a network from RNAi microarray time series data. They infer both feedforward and feedback loops from estimated time delays and also capture the stochastic nature of signaling processes.

7.2 Future directions

This thesis has discussed the potential usefulness of DNEMs to analyze genomic perturbation data. However there are limitations of the current representation and learning approaches that need further investigation. The work in this dissertation can be extended in many directions.

7.2.1 Combinatorial perturbations

NEMs handle data from single knock down experiments. Recall that the early ESC development involves key TFs like *Cdx2* whose induction can trigger stem cell differentiation as well as the knock-down of other important TFs like $Nanog$, $Sox2$, and $Oct4$ (18). Thus it is possible to have both knock down and knock in experimental data generated from the same biological model. Furthermore, recall that in the context of NEMs the first blocked signal wins in an AND gate interaction between S-genes. The AND becomes an OR since only one of the incoming signals is needed to break signal flow. This scenario changes when dealing with knock-in data. A downstream S-gene gets activated only when all its parents are activated in an AND situation. Hence an OR-NEM is equivalent to an AND-knock-in NEM. In general, based on the type of perturbed data, the topology Φ together with the set of boolean functions F defines a deterministic Boolean network on the set of S-genes S . This corresponds generally to fewer perturbation schemes on S. Of course we would have to deal with the situation where several hypotheses with different perturbation schemes produce identical data. The challenge would be to find all equivalence classes under different experimental conditions such as single knock-downs, single knock-ins, or even a combination of both knock-in and knock-down experiments involving different boolean functions. Furthermore, more sophisticated perturbation schemes have to be developed, which encode predictions both from single-gene and multi-gene knock-outs and knock-ins. Since the number of possible multiple knock-outs and knock-ins increases exponentially, we need tools to choose the most informative experiments. Ultimately, reconstructing very large informative networks from perturbation data still remain an open area for interesting research.

7.2.2 NEMs and drug interventions

RNAi has become a method of choice for key steps in the development of therapeutic agents, from target discovery and validation to the analysis of the mechanisms of action of small molecules. In the framework of NEMs or DNEMs if we replace the S-genes with drug intervention schemes, we may be able to identify suitable drug targets and their genetic models in cancer therapy by inferring which genes when stimulated with a drug, promote cell suicide in tumor cells, but not in normal body tissue. Furthermore, with the availability of large drug interventional databases showing changes in gene expression profiles across the entire human genome as well as information on gene ontology we could use NEMs to identify cluster of drugs with underlying similar molecular and phenotypic properties. Work in this direction still needs to be done.

7.3 Food for thought

A very optimistic Uri Alon (7) wrote that " There is no a priori reason that immensely complex biological systems would be understandable. But despite the fact that biological networks evolved to function and not to be comprehensible, simplifying principles can be found that make biological design understandable to us". I believe that, a first step to understand the complex inner working of a cell is by breaking it into simpler comprehensive circuits. This thesis explored one of the possible ways of inferring the dynamics of complex biological systems from gene expression data. The most striking feature of the early stem cell differentiation model is the high frequency of both FFLs and FBLs. This opens up a wide spectrum of pathway hypotheses, raising the question of why evolution has conserved these simple modules in such a complex network topology.

8

Appendix

8.1 Derivation of the Probability density of signal propagation along a linear path

Let us first consider a fixed linear path g in Φ, which connects the S-gene S_i with the E-gene E_k:

$$S_i \xrightarrow{k_1} S_{j_1} \cdots \xrightarrow{k_{q-1}} S_{j_{q-1}} \xrightarrow{k_q} E_k,$$

We want to calculate the probability that the signal has reached E_k at time point t. In general Let $X_0 \xrightarrow{T_1} X_1 \xrightarrow{T_2} \cdots \xrightarrow{T_n} X_n$ be a linear path with edge weights T_j. We denote 1 as active state and 0 otherwise. If $X_0 = 1$ at timepoint 0, then the probability that the signal has reached X_n at timepoint t is

$$\begin{aligned} f(t) &= P(X_n = 1|t) = P(\sum_{j=1}^{n} T_j < t) \\ &= \int_{s=0}^{t} p(\sum_{j=1}^{n} T_j = u)ds \\ &= \int_{s=0}^{t} \left(\int_{\sum_{j=1}^{n} t_j = u} p(T_j = t_j)dt_1 dt_2 ... dt_n \right) du \\ &= \int_{s=0}^{t} (\psi_1 * \psi_2 ... * \psi_n)(u)du \qquad (8.1) \end{aligned}$$

with the density functions $\psi_t = k_j e^{(-k_j t)} \delta_{t>0}$. The integration in Equation 8.1 can be solved using the Laplace transform $f \rightarrow \hat{f} : (s \mapsto \int_{-\infty}^{\infty} e^{-st} f(t)dt)$. Using well known rules for the Laplace transform, we obtain

$$\hat{f} = \frac{1}{s}\prod_{j=1}^{n}\hat{\psi}_j(s) = \frac{1}{s}\prod_{j=1}^{n}\frac{k_j}{s+k_j} \tag{8.2}$$

We use partial fractions expansion to resolve the last term in Equation 8.2. Define $P(s)$ $= \prod_{j=1}^{n}(s+k_j)$ and let the Lagrange polynomials $P_j(s) = \frac{s+k_j}{k_j-k_i}$. Note that $P_j(-k_r)$ $= \delta_{r=i}$. Hence $P_j(s)|j = 1, ..., n$ form the basis of the vector space of polynomials of degree at most $n-1$, and the constant polynomial 1 has the representation

$$1 = \sum_{j=1}^{n} P_j(s)$$

Dividing by $P(s)$ we obtain

$$\prod_{j=1}^{n}\frac{1}{s+k_j} = \frac{1}{P(s)} = \sum_{j=1}^{n}\frac{P_j(s)}{P(s)} = \sum_{j=1}^{n}\left\{\prod_{i\neq j}\frac{1}{k_j-k_i}\right\}\frac{1}{s+k_j} \tag{8.3}$$

from which we deduce:

$$\begin{aligned}\hat{f} &= \frac{1}{s}\prod_{j=1}^{n}\frac{k_j}{s+k_j} \overset{7.3}{=} \sum_{j=1}^{n}\left\{\prod_{i\neq j}\frac{1}{k_j-k_i}\right\}\frac{1}{s+k_j}\\ &= \frac{1}{s}\sum_{j=1}^{n}\left\{\prod_{i\neq j}\frac{k_j}{k_j-k_i}\right\}\frac{k_j}{s+k_j} = \sum_{j=1}^{n}Q_j\frac{k_j}{s(s+k_j)} \end{aligned}\tag{8.4}$$

with $Q_j = \prod_{i\neq j}\frac{k_j}{k_j-k_i}$. Using the inverse Laplace transform $\hat{f}^{-1}$, we finally obtain

$$f(t) = \hat{f}^{-1} = \sum_{j=1}^{n}Q_j\left[\frac{k_j}{s(s+k_j)}\right]^{-1}(t) = \sum_{j=1}^{n}Q_j(1-e^{-k_jt}) \tag{8.5}$$

Equation 8.5 provides a closed form expression for the signal probability density of time along a linear path with exponentially distributed delay times on the edges.

References

[1] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell. 5th edition.* Garland Science, New York, 2008. iii, 2, 3, 5, 6, 10

[2] Charles J Vaske, Carrie House, Truong Luu, Bryan Frank, Chen-Hsiang Yeang, Norman H Lee, and Joshua M Stuart. **A factor graph nested effects model to identify networks from genetic perturbations.** *PLoS Comput Biol*, **5**(1):e1000274, Jan 2009. iii, 23, 24, 26, 27

[3] Uri Alon. **Network motifs: theory and experimental approaches.** *Nat Rev Genet*, **8**(6):450–461, Jun 2007. iii, 9, 10, 11

[4] Hitoshi Niwa. **How is pluripotency determined and maintained?** *Development*, **134**(4):635–646, Feb 2007. iii, 3, 5

[5] H. Wajant. **The Fas signaling pathway: more than a paradigm**. *Science*, **296 (5573)**:16356, 2002. 3

[6] Nika N Danial and Stanley J Korsmeyer. **Cell death: critical control points.** *Cell*, **116**(2):205–219, Jan 2004. 3

[7] Uri Alon. *An introduction to systems biology : Design Principles of biological circuits.* CHAPMAN & HALL/CRC, 2007. 3, 8, 9, 10, 96

[8] A. K. TARKOWSKI. **Experiments on the development of isolated blastomers of mouse eggs.** *Nature*, **184**:1286–1287, Oct 1959. 3

[9] Davor Solter. **From teratocarcinomas to embryonic stem cells and beyond: a history of embryonic stem cell research.** *Nat Rev Genet*, **7**(4):319–327, Apr 2006. 4

[10] Y. Suda, M. Suzuki, Y. Ikawa, and S. Aizawa. **Mouse embryonic stem cells exhibit indefinite proliferative potential.** *J Cell Physiol*, **133**(1):197–201, Oct 1987. 4

[11] H. Niwa, J. Miyazaki, and A. G. Smith. **Quantitative expression of Oct-3/4 defines differentiation, dedifferentiation or self-renewal of ES cells.** *Nat Genet*, **24**(4):372–376, Apr 2000. 4

[12] Junji Fujikura, Eiji Yamato, Shigenobu Yonemura, Kiminori Hosoda, Shinji Masui, Kazuwa Nakao, Jun ichi Miyazaki Ji, and Hitoshi Niwa. **Differentiation of embryonic stem cells is induced by GATA factors.** *Genes Dev*, **16**(7):784–789, Apr 2002. 4

[13] Kaoru Mitsui, Yoshimi Tokuzawa, Hiroaki Itoh, Kohichi Segawa, Mirei Murakami, Kazutoshi Takahashi, Masayoshi Maruyama, Mitsuyo Maeda, and Shinya Yamanaka. **The homeoprotein Nanog is required for maintenance of pluripotency in mouse epiblast and ES cells.** *Cell*, **113**(5):631–642, May 2003. 4

[14] Ian Chambers, Douglas Colby, Morag Robertson, Jennifer Nichols, Sonia Lee, Susan Tweedie, and Austin Smith. **Functional expression cloning of Nanog, a pluripotency sustaining factor in embryonic stem cells.** *Cell*, **113**(5):643–655, May 2003. 4

[15] Laurie A Boyer, Tong Ihn Lee, Megan F Cole, Sarah E Johnstone, Stuart S Levine, Jacob P Zucker, Matthew G Guenther, Roshan M Kumar, Heather L Murray, Richard G Jenner, David K Gifford, Douglas A Melton, Rudolf Jaenisch, and Richard A Young. **Core transcriptional regulatory circuitry in human embryonic stem cells.** *Cell*, **122**(6):947–956, Sep 2005. 4

[16] Hitoshi Niwa, Yayoi Toyooka, Daisuke Shimosato, Dan Strumpf, Kadue Takahashi, Rika Yagi, and Janet Rossant. **Interaction between Oct3/4 and Cdx2 determines trophectoderm differentiation.** *Cell*, **123**(5):917–929, Dec 2005. 4

[17] Natalia Ivanova, Radu Dobrin, Rong Lu, Iulia Kotenko, John Levorse, Christina DeCoste, Xenia Schafer, Yi Lun, and Ihor R Lemischka. **Dissecting self-renewal in stem cells with RNA interference.** *Nature*, **442**(7102):533–538, Aug 2006. 4, 61, 79, 90

[18] Akira Nishiyama, Li Xin, Alexei A Sharov, Marshall Thomas, Gregory Mowrer, Emily Meyers, Yulan Piao, Samir Mehta, Sarah Yee, Yuhki Nakatake, Carole Stagg, Lioudmila Sharova, Lina S Correa-Cerro, Uwem Bassey, Hien Hoang, Eugene Kim, Richard Tapnio, Yong Qian, Dawood Dudekula, Michal Zalzman, Manxiang Li, Geppino Falco, Hsih-Te Yang, Sung-Lim Lee, Manuela Monti, Ilaria Stanghellini, Md Nurul Islam, Ramaiah Nagaraja, Ilya Goldberg, Weidong Wang, Dan L Longo, David Schlessinger, and Minoru S H Ko. **Uncovering early response of gene regulatory**

networks in ESCs by systematic induction of transcription factors. *Cell Stem Cell*, **5**(4):420–433, Oct 2009. 4, 5, 83, 95

[19] Ian Chambers, Jose Silva, Douglas Colby, Jennifer Nichols, Bianca Nijmeijer, Morag Robertson, Jan Vrana, Ken Jones, Lars Grotewold, and Austin Smith. **Nanog safeguards pluripotency and mediates germline development.** *Nature*, **450**(7173):1230–1234, Dec 2007. 4

[20] Shinji Masui, Yuhki Nakatake, Yayoi Toyooka, Daisuke Shimosato, Rika Yagi, Kazue Takahashi, Hitoshi Okochi, Akihiko Okuda, Ryo Matoba, Alexei A Sharov, Minoru S H Ko, and Hitoshi Niwa. **Pluripotency governed by Sox2 via regulation of Oct3/4 expression in mouse embryonic stem cells.** *Nat Cell Biol*, **9**(6):625–635, Jun 2007. 4

[21] Laurie A Boyer, Tong Ihn Lee, Megan F Cole, Sarah E Johnstone, Stuart S Levine, Jacob P Zucker, Matthew G Guenther, Roshan M Kumar, Heather L Murray, Richard G Jenner, David K Gifford, Douglas A Melton, Rudolf Jaenisch, and Richard A Young. **Core transcriptional regulatory circuitry in human embryonic stem cells.** *Cell*, **122**(6):947–956, Sep 2005. 4

[22] David J Rodda, Joon-Lin Chew, Leng-Hiong Lim, Yuin-Han Loh, Bei Wang, Huck-Hui Ng, and Paul Robson. **Transcriptional regulation of nanog by OCT4 and SOX2.** *J Biol Chem*, **280**(26):24731–24737, Jul 2005. 4

[23] F. Beck, T. Erler, A. Russell, and R. James. **Expression of Cdx-2 in the mouse embryo and placenta: possible role in patterning of the extra-embryonic membranes.** *Dev Dyn*, **204**(3):219–227, Nov 1995. 4

[24] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. **Causal protein-signaling networks derived from multiparameter single-cell data.** *Science*, **308**(5721):523–529, Apr 2005. 6, 8

[25] Y. Yamanishi, J-P. Vert, and M. Kanehisa. **Protein network inference from multiple genomic data: a supervised approach.** *Bioinformatics*, **20 Suppl 1**:i363–i370, Aug 2004. 6

[26] Seiya Imoto, Takao Goto, and Satoru Miyano. **Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression.** *Pac Symp Biocomput*, pages 175–186, 2002. 6

[27] Joshua M Stuart, Eran Segal, Daphne Koller, and Stuart K Kim. **A gene-coexpression network for global discovery of conserved genetic modules.** *Science*, **302**(5643):249–255, Oct 2003. 8

[28] Anja Wille, Philip Zimmermann, Eva Vranov, Andreas Frholz, Oliver Laule, Stefan Bleuler, Lars Hennig, Amela Prelic, Peter von Rohr, Lothar Thiele, Eckart Zitzler, Wilhelm Gruissem, and Peter Bhlmann. **Sparse graphical Gaussian modeling of the isoprenoid gene network in Arabidopsis thaliana.** *Genome Biol*, **5**(11):R92, 2004. 8

[29] Juliane Schaefer and Korbinian Strimmer. **An empirical Bayes approach to inferring large-scale gene association networks.** *Bioinformatics*, **21**(6):754–764, Mar 2005. 8

[30] Katia Basso, Adam A Margolin, Gustavo Stolovitzky, Ulf Klein, Riccardo Dalla-Favera, and Andrea Califano. **Reverse engineering of regulatory networks in human B cells.** *Nat Genet*, **37**(4):382–390, Apr 2005. 8

[31] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. **Using Bayesian networks to analyze expression data.** *J Comput Biol*, **7**(3-4):601–620, 2000. 8, 12

[32] S. Mian K. Murphy. **Modelling gene expression data using dynamic Bayesian networks**. Technical report, Computer Science Division, University of California, Berkeley, CA, 1999. 8

[33] Dirk Husmeier. **Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks.** *Bioinformatics*, **19**(17):2271–2282, Nov 2003. 8

[34] S. A. Kauffman. *Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press. Technical monograph, 1993. 8

[35] M. Aldana. **Boolean dynamics of networks with scale-free topology.** *Physica D*, **185**:45–66, 2003. 8

[36] Ilya Shmulevich and Edward R. Dougherty. *Probabilistic Boolean Networks. The Modeling and Control of Gene Regulatory Networks:*. Society for Industrial and Applied Mathematics, Philadelphia, 2010. 8

[37] Minh Quach, Nicolas Brunel, and Florence d'Alch Buc. **Estimating parameters and hidden variables in non-linear state-space models based on ODEs for biological networks inference.** *Bioinformatics*, **23**(23):3209–3216, Dec 2007. 8

[38] Edda Klipp and Wolfram Liebermeister. **Mathematical modeling of intracellular signaling pathways.** *BMC Neurosci*, **7 Suppl 1**:S10, 2006. 8

[39] Henning Schmidt and Mats Jirstrand. **Systems Biology Toolbox for MATLAB: a computational platform for research in systems biology.** *Bioinformatics*, **22**(4):514–515, Feb 2006. 8

[40] Jonathan M Raser and Erin K O'Shea. **Control of stochasticity in eukaryotic gene expression.** *Science*, **304**(5678):1811–1814, Jun 2004. 8

[41] Nitzan Rosenfeld, Jonathan W Young, Uri Alon, Peter S Swain, and Michael B Elowitz. **Gene regulation at the single-cell level.** *Science*, **307**(5717):1962–1965, Mar 2005. 8

[42] M. A. Savageau. **Michaelis-Menten mechanism reconsidered: implications of fractal kinetics.** *J Theor Biol*, **176**(1):115–124, Sep 1995. 8

[43] Florian Markowetz and Rainer Spang. **Inferring cellular networks–a review.** *BMC Bioinformatics*, **8 Suppl 6**:S5, 2007. 8

[44] Adriano V Werhli, Marco Grzegorczyk, and Dirk Husmeier. **Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical gaussian models and bayesian networks.** *Bioinformatics*, **22**(20):2523–2531, Oct 2006. 8

[45] *Evaluating the effect of perturbations in reconstructing network topologies. Proceedings of the 3rd International Workshop on Distributed Statistical Computing, March 2022, 2003, Vienna, Austria.*, 2003. 8

[46] J Rung, T Schlitt, A Brazma, K Freivalds, and J Vilo. **Building and analyzing genomewide gene disruption networks.** *Bioinformatics*, **18**:202–210, 2002. 8

[47] Andreas Wagner. **Estimating coarse gene network structure from large-scale gene perturbation data.** *Genome Res*, **12**(2):309–315, Feb 2002. 8

[48] D. Pe'er, A. Regev, G. Elidan, and N. Friedman. **Inferring subnetworks from perturbed expression profiles.** *Bioinformatics*, **17 Suppl 1**:S215–S224, 2001. 8

[49] Florian Markowetz, Jacques Bloch, and Rainer Spang. **Non-transcriptional pathway features reconstructed from secondary effects of RNA interference.** *Bioinformatics*, **21**(21):4026–4032, Nov 2005. 8, 9, 15, 16, 17, 18, 23, 24, 27

[50] Chen-Hsiang Yeang, Trey Ideker, and Tommi Jaakkola. **Physical network models.** *J Comput Biol*, **11**(2-3):243–262, 2004. 9

[51] Shai S Shen-Orr, Ron Milo, Shmoolik Mangan, and Uri Alon. **Network motifs in the transcriptional regulation network of Escherichia coli.** *Nat Genet*, **31**(1):64–68, May 2002. 9

[52] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. **Network motifs: simple building blocks of complex networks.** *Science*, **298**(5594):824–827, Oct 2002. 9

[53] A. Becskei and L. Serrano. **Engineering stability in gene networks by autoregulation.** *Nature*, **405**(6786):590–593, Jun 2000. 12

[54] Francisco M Camas, Jess Blzquez, and Juan F Poyatos. **Autogenous and nonautogenous control of response in a genetic network.** *Proc Natl Acad Sci U S A*, **103**(34):12718–12723, Aug 2006. 12

[55] Nitzan Rosenfeld, Michael B Elowitz, and Uri Alon. **Negative autoregulation speeds the response times of transcription networks.** *J Mol Biol*, **323**(5):785–793, Nov 2002. 12

[56] Yusuke T Maeda and Masaki Sano. **Regulatory dynamics of synthetic gene networks with positive feedback.** *J Mol Biol*, **359**(4):1107–1124, Jun 2006. 12

[57] Florian Markowetz, Dennis Kostka, Olga G Troyanskaya, and Rainer Spang. **Nested effects models for high-dimensional phenotyping screens.** *Bioinformatics*, **23**(13):i305–i312, Jul 2007. 19, 27, 28, 44, 50, 55, 63

[58] Holger Froehlich, Mark Fellmann, Holger Sueltmann, Annemarie Poustka, and Tim Beissbarth. **Large scale statistical inference of signaling pathways from RNAi and microarray data.** *BMC Bioinformatics*, **8**:386, 2007. 19, 27, 28, 89

[59] Achim Tresch and Florian Markowetz. **Structure learning in Nested Effects Models.** *Stat Appl Genet Mol Biol*, **7**(1):Article9, 2008. 20, 21, 22, 23, 27, 28, 45, 52, 55, 86

[60] Cordula Zeller, Holger Froehlich, and AchimTresch. **A Bayesian Network View on Nested EffectsModels**. *EURASIP Journal on Bioinformatics and Systems Biology*, **2009**, 2009. 21

[61] *Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning. (UCLA Technical Report CSD-850017). Proceedings of the 7th Conference of the Cognitive Science Society, University of California, Irvine, CA. pp. 329334.*, 1985. 21

[62] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, San Francisco, CA, 1988. 21

[63] Kschischang, Frey, and Loeliger. **Factor graphs and the sum-product algorithm**. *IEEE Transactions on Information Theory*, **47**, 2001. 24, 27

[64] Brendan J. Frey and David J. C. MacKay. **A Revolution: Belief Propagation in Graphs With Cycles**. In *In Neural Information Processing Systems*, pages 479–485. MIT Press, 1997. 27

[65] Brendan J Frey and Delbert Dueck. **Clustering by passing messages between data points.** *Science*, **315**(5814):972–976, Feb 2007. 27

[66] David J.C. MacKay, David J. C. Mackay, Radford M. Neal, and Radford M. Neal. *Good Codes based on Very Sparse Matrices*. Springer, 1995. 27

[67] Juby Jacob, Marcel Jentsch, Dennis Kostka, Stefan Bentink, and Rainer Spang. **Detecting hierarchical structure in molecular characteristics of disease using transitive approximations of directed graphs.** *Bioinformatics*, **24**(7):995–1001, Apr 2008. 28

[68] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis(2nd Edition)*. Chapman & Hall/CRC, 2004. 31, 36

[69] W. K. Hastings. **Monte Carlo Sampling Methods Using Markov Chains and Their Applications**. *Biometrika*, **57(1)**:97–109, 1970. 31, 35

[70] S. Geman and D. B. Geman. **Stochastic relaxation, Gibbs distribution, and Bayes restoration of images.** *IEEE Transactions on pattern recognition and artificial intelligence*, **6**:721–741, 1984. 31, 32

[71] Martin A. Tanner and Wing Hung Wong. **The Calculation of Posterior Distributions by Data Augmentation**. *Journal of the American Statistical Association*, **82**:528–540, 1987. 31

[72] Alan E. Gelfand and Adrian F. M. Smith. **Sampling-Based Approaches to Calculating Marginal Densities**. *Journal of the American Statistical Association*, **85**:398–409, 1990. 31, 34

[73] A. P. Dempster, N. M. Laird, and D. B. Rubin. **Maximum Likelihood from Incomplete Data via the EM Algorithm**. *Journal of the Royal Statistical Society*, **Series B (Methodological) 39 (1)**:1–38, 1977. 31

[74] Sujit K. Sahu and Gareth O. Roberts. **On Convergence of the EM Algorithm and the Gibbs Sampler**. *Statistics and Computing*, **9**:9–55, 1998. 32

[75] Charles E. Lawrence, Stephen F. Altschul, Mark S. Boguski, Jun S. Liu, Andrew F. Neuwald, and John C. Wootton. **Detecting Subtle Sequence Signals: A Gibbs Sampling Strategy for Multiple Alignment**. *Science*, **262**:208–214, 1993. 32

[76] Qizheng Sheng, Yves Moreau, and Bart De Moor. **Biclustering microarray data by Gibbs sampling**. *Bioinformatics*, **19**:196–205, 2003. 32

[77] Jun S. Liu, Andrew F. Neuwald, and Charles E. Lawrence. **Bayesian Models for Multiple Local Sequence Alignment and Gibbs Sampling Strategies**. *Journal of the American Statistical Association*, **90**:1156–1170, 1995. 32

[78] Modan K Das and Ho-Kwok Dai. **A survey of DNA motif finding algorithms.** *BMC Bioinformatics*, **8 Suppl 7**:S21, 2007. 32

[79] George Casella and Edward I. George. **Explaining the Gibbs Sampler**. *The American Statistician*, **46**:167–174, 1992. 33

[80] R. E. Caflisch. *Monte Carlo and quasi-Monte Carlo methods*. Cambridge University Press, 1998. 34

[81] D. Blackwell. **Conditional expectation and unbiased sequential estimation**. *Annals of Mathematical Statistics*, **18(1)**:105–110, 1947. 34

[82] *A tutorial on hidden Markov models and selected applications in speech recognition.*, 1989. 35

[83] Jun S. Liu. **The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem**. *Journal of the American Statistical Association*, **84**, 1994. 35

[84] Radford M. Neal. **Suppressing Random Walks in Markov Chain Monte Carlo Using Ordered Overrelaxation.Technical Report University of Toronto, Department of Statistics 9508**. 1995. 35

[85] Radford M. Neal. **Slice Sampling**. *Annals of Statistics*, **31(3)**:705–767, 2003. 35

[86] N. Metropolis, A..W.Rosenbluth, M.N. Rosenbluth, A.H.Teller, and E. Teller. **Equations of State Calculations by Fast Computing Machines**. *Journal of Chemical Physics*, **21(6)**:1087–1092, 1953. 35

[87] Peter McCullagh and John Nelder. *Generalized Linear Models(2nd Edition)*. Chapman and Hall/CRC, 1989. 36

[88] A Gelman and D. B.Rubin. **Inference from Iterative Simulation Using Multiple Sequences**. *Statistical Science*, **7**:457–472, 1992. 37

[89] S. P. Brooks and A. Gelman. **General Methods for Monitoring Convergence of Iterative Simulations**. *Journal of Computational and Graphical Statistics*, **7**:434–455, 1997. 37

[90] John Geweke. **Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments**. In *IN BAYESIAN STATISTICS*, pages 169–193. University Press, 1992. 38

[91] A. E. Raftery and S. M. Lewis. **One Long Run with Diagnostics: Implementation Strategies for Markov Chain Monte Carlo**. *Statistical Science*, **7**:493–497, 1992. 38

[92] A. E. Raftery and S. M. Lewis. **The Number of Iterations, Convergence Diagnostics and Generic Metropolis Algorithms**. In *In Practical Markov Chain Monte Carlo (W.R. Gilks, D.J. Spiegelhalter and*, pages 115–130. Chapman and Hall, 1995. 38

[93] P. Heidelberger and P. D. Welch. **A Spectral Method for Confidence Interval Generation and Run Length Control in Simulations**. *Communication of the ACM*, **24**:233–245, 1981. 39

[94] Philip Heidelberger and Peter D. Welch. **Simulation Run Length Control in the Presence of an Initial Transient**. *Operations Research*, **31**:1109–1144, 1983. 39

[95] T.W. Anderson. **On the Distribution of the Two-Sample Cramer-von Mises Criterion**. *The Annals of Mathematical Statistics*, **33**:1148–1159, 1962. 39

[96] Nicholas M Luscombe, M. Madan Babu, Haiyuan Yu, Michael Snyder, Sarah A Teichmann, and Mark Gerstein. **Genomic analysis of regulatory network dynamics reveals large topological changes.** *Nature*, **431**(7006):308–312, Sep 2004. 40

[97] Marcel Ovidiu Vlad, Federico Moran, Masa Tsuchiya, L. Luca Cavalli-Sforza, Peter J Oefner, and John Ross. **Neutrality condition and response law for nonlinear reaction-diffusion equations, with application to population genetics.** *Phys Rev E Stat Nonlin Soft Matter Phys*, **65**(6 Pt 1):061110: 1–17, Jun 2002. 44

[98] Marcel O Vlad, Adam Arkin, and John Ross. **Response experiments for nonlinear systems with application to reaction kinetics and genetics.** *Proc Natl Acad Sci U S A*, **101**(19):7223–7228, May 2004. 44

[99] P.G Moschopoulos. **The distribution of sum of independent gamma random variables**. *Annals of the Institute of Statistical Mathematics*, **37**:541–544, 1984. 46

[100] H. Jasiulewicz and W. Kordecki. **Convolutions of Erlang and of Pascal distributions with applications to reliability**. *Demonstratio Mathematica*, **36(1)**:231–238, 2003. 47

[101] S.M. Ross. *Introduction to Probability Models sixth edition.* Academic Press, San Diego, CA, 1997. 47

[102] V.G. Kulkarni. **Shortest paths in networks with exponentially distributed arc lengths**. *Networks*, **16**:255–274, 1986. 48

[103] Holger Froehlich, Tim Beissbarth, Achim Tresch, Dennis Kostka, Juby Jacob, Rainer Spang, and F. Markowetz. **Analyzing gene perturbation screens with nested effects models in R and bioconductor.** *Bioinformatics*, **24**(21):2549–2550, Nov 2008. 52, 55

[104] Gordon K Smyth. **Linear models and empirical bayes methods for assessing differential expression in microarray experiments.** *Stat Appl Genet Mol Biol*, **3**:Article3, 2004. 52

[105] DJ Spiegelhalter, NG Best, BP Carlin, and A van der Linde. **Bayesian measures of model complexity and fit**. *J.R. Statist. Soc.*, **64(4)**:583–616, 2002. 55

[106] David L. Applegate, Robert E. Bixby, and William J. Cook. *The Traveling Salesman Problem: A Computational Study.* Princeton University Press, 2006. 72

[107] Holger Froehlich, Paurush Praveen, and Achim Tresch. **Fast and efficient dynamic nested effects models.** *Bioinformatics*, **27**(2):238–244, Jan 2011. 85

[108] Zoubin Ghahramani. **Learning Dynamic Bayesian Networks: Lecture Notes In Computer Science**. **1387**:168–197. 86

[109] Gideon E. Schwarz. **Estimating the dimension of a model**. *Annals of Statistics*, **6 (2)**:461–464, 1978. 89

[110] Sundar Dorai-Raj. *R Package binom: Binomial Confidence Intervals For Several Parameterizations*, 2009. 90

Curriculum Vitae

Address

Institute for Functional Genomics
Department of Statistical Bioinformattics
University of Regensburg, Josef Engertstr. 9
93053 Regensburg, Germany.Tel: 0049 (0)941 943 1584
Email:benedict.anchang@klinik.uni-regensburg.de

Education

June 2007 - Present	**University of Regensburg**
PhD candidate in Bioinformatics	Advisor: Prof. Rainer Spang
Oct 2005 - Nov 2006	**Transnational University Limburg**
Master of Science in Biostatistics	Advisor: Prof. Ziv Shkedy
Oct 2004 - Sept 2005	**University of Hasselt**
Master of Science in Applied Statistics	Advisor: Prof. Herbert Thijs
Oct 1998 - July 2002	**University of Buea, Cameroon**
Bachelor of Science in Mathematics	Minor: Computer Science

Working Experience

June 2007-present	**Institute of Functional Genomics**
Research assistant	University of Regensburg, Germany
July 2006- Sept 2006	**National Institute of Public Health**
Student Intern	3720 BA Bilthoven Netherlands

International Conferences

Sep 26-29th 2010	**ECCB Ghent, Belgium**
July 13th 2010	**ISMB Boston, USA**
Dec 1-4th 2009	**ISCB Bamako, Mali**
November 1-4th 2009	**IEEE Bethesda, USA**

Declaration

I herewith declare that I have produced this thesis without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This dissertation has not previously been presented in identical or similar form to any other German or foreign examination board.

The thesis work was conducted from 1-06-2007 to 31-09-2011 under the supervision of Prof.Dr. Wolfram Gronwald and Prof.Dr. Rainer Spang at the Institute of Functional Genomics, University of Regensburg, Germany.

REGENSBURG,