Published in final edited form as:

IEEE Signal Process Mag. 2009 January 1; 26(1): 76–97. doi:10.1109/MSP.2008.930647.

A Survey of Statistical Models for Reverse Engineering Gene Regulatory Networks

Yufei Huang,

Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, TX 78249-0669, yufei.huang@utsa.edu

Isabel M. Tienda-Luna, and

Department of Electronics and Computer Science, University of Granada, Spain

Yufeng Wang

Department of Biology, University of Texas at San Antonio, San Antonio, TX

Abstract

Statistical models for reverse engineering gene regulatory networks are surveyed in this article. To provide readers with a system-level view of the modeling issues in this research, a graphical modeling framework is proposed. This framework serves as the scaffolding on which the review of different models can be systematically assembled. Based on the framework, we review many existing models for many aspects of gene regulation; the pros and cons of each model are discussed. In addition, network inference algorithms are also surveyed under the graphical modeling framework by the categories of point solutions and probabilistic solutions and the connections and differences among the algorithms are provided. This survey has the potential to elucidate the development and future of reverse engineering GRNs and bring statistical signal processing closer to the core of this research.

I. Introduction

Recent advances in high-throughput biological technologies typified by gene sequencing, DNA microarray, and proteomics experiments have unleashed a torrent of different types of data that measure DNA sequence information, mRNA transcript quantities, protein/peptide expressions, protein-protein interactions, and many others. The proliferation of disparate data has caused researchers to attempt to understand, using computational approaches, complex biological processes at a system level, resulting in a nascent field known as computational systems biology (CSB). CSB offers great promise for a *global* understanding of the expressions, interactions, modifications, and regulation of cellular networks.

A cellular network is a collection of molecular interactions. One type of the cellular networks is the gene regulatory network (GRN). A GRN is a network representing regulations between genes in a cell and is the regulatory circuit linking proteins and targets. Genes are nodes in this network and edges are regulatory relationships between genes. GRNs are of great importance in understanding cellular system structure, system dynamics, control mechanisms, and design principles. A GRN can be also viewed as an input-output device; the inputs are signals, signaling pathways, transcriptional factors (TFs), etc., and the outputs are gene expression levels or the amount of mRNA produced by genes. Presently, an active research area in computational systems biology is to uncover or reverse engineer the GRNs based on the inputs and outputs of the networks [1]. Reverse engineering GRNs is the focus of this article.

The key issues in reverse engineering GRNs, and in CSB at large, revolve around system modeling and computational inference. Accurate modeling of gene regulation at the system level is a crucial step. System-level modeling of gene regulation requires precise mathematical descriptions of various aspects of biological mechanisms contributing to the regulatory relationships in a network and of experimental conditions affecting the outcome of the network. Such models reveal connections between the regulatory networks and data, providing clues for uncovering the network from data. Modeling has been a vital topic in the GRNs research and a large body of work exists including, most noticeably, (probabilistic) Boolean networks [2], Bayesian networks [1], signed directed graphs (SDG) [3], and differential equations [4]. Our focus in this article is statistical models appropriate for inferential computations used in reverse engineering GRNs, rather than those for network simulations.

In the process of statistical modeling for GRNs, four aspects must be carefully scrutinized: accuracy, robustness, adaptivity, and a system-level view. First, the models should be as faithful as possible to the underlying biology and experimental designs. This will call for employing increasingly complex models along with more advanced inference techniques. Additionally, we need to consider the robustness of a model, a feature that leads to reliable inference. On the one hand, this requires us to trade off model complexities for the fidelity of inference; on the other hand, robust modeling techniques are highly desirable and data integration becomes indispensable. Moreover, the process is iterative and dynamic; the computational results will guide subsequent experiments and the new experimental results will in turn be integrated into computations to improve the inference results. Consequently, the models must be adaptive, i.e., we must be able to effectively modify and extend their current structures to reflect the changes in knowledge. Finally, systems biology research deserves system-level modeling. System-level modeling, as opposed to localized modeling, should include sufficient components in its modeling assay so as to present a global view of gene regulation.

In this article, we survey existing statistical models for uncovering GRNs, through which we aim to stress previously mentioned aspects of modeling and shed light on the continuing research effort. Excellent surveys on general computational issues for GRNs exist in the literature [4], [1], [2]. However, there are many important issues yet to be addressed. While the articles such as [1] targeted static experiments with linear and/or discrete models, limited attention has been given to continuous modeling of the dynamics and nonlinearity of gene regulation. Survey articles such as [4] provide a balanced review of both simulation and computation models with limited discussion of computational models. Some of these review articles fail to reflect the current developments, such as data integration, which is not surveyed in any of the above articles. Finally, all of these articles discuss different modeling issues for different problems. Even though they survey each topic in detail, they provide no discussion as to how different models are compared and what the connections among them are. As a result, they fail to present a system-level landscape to readers. These problems make it difficult for readers, especially those from signal processing communities, to recognize the prospects of this research.

The goal of this article is to provide readers from signal processing and/or general computational backgrounds with an in-depth, comprehensive, and up-to-date survey on modeling and related inference issues in GRN research. We review as many models as possible, as well as present a picture coherent with the nature of this research and a picture of modeling that reflects the above requirements. We also address shortcomings of the current survey papers, in an attempt to gain a system-level view of the topic. To reach this goal, we present the survey of models and inference under a graphical modeling framework (GMF), which consists of connected layers, each of which describes a different aspect of the gene regulatory system. Depending on its function, each existing model will be categorized and reviewed at its corresponding layer, and in particular, the survey of the modeling of the prior data will be

carried out within the GMF. Overall, the framework serves as a blueprint, on which different existing models can be put together to obtain a system-level view on the modeling of GRNs. The framework also presents tangible connections among different existing models, making the comparisons straightforward. Under the GMF, network inference algorithms will be also reviewed in a systematic fashion. Since graphical modeling is a familiar tool for system modeling in signal processing and machine learning, it is reasonable to expect the survey, based on GMF, to easily resonate with researchers in the signal processing community and beyond.

The rest of the article is organized as follows: In section II, we briefly review the biological background of gene regulation and the different types of data often used in this research. In section III, a graphical modeling framework is presented. Based on the framework, different existing models for different datasets and biological processes are surveyed in section IV. Inference issues and approaches are reviewed in section V. Concluding remarks are provided in section VII.

II. Background

A. Gene Regulation and Regulatory Networks

Gene regulation is the cellular control of the abundance and timing of the functional product of a gene [5]. Gene expression may change in response to physical signals from the environment, interactions across species, and signals within an organism/tissue/cell. Regulatory mechanisms operate at various levels: through on-off switches of DNA-RNA transcription, changes of stability, or translations of mRNA, or alterations in protein activity through post-translational modification.

Figure 1 shows a schematic of gene regulation that involves signal transduction. Signals or stimuli such as chemical gradients carry instructions for synthesizing proteins; the receptors localized outside the cell membrane often constitute channels which allow signals to be passed, for example, in the form of small ion movement. These ion movements result in changes in the electrical potential of the cells and amplify the signal in a cascade mode along the cell. This process of relaying the instruction messages through various enzymes and messenger molecules is called “signal transduction.” After a message is passed into the nucleus, the activation of a protein called transcription factor (TF) is initiated. TF binds directly to a specific upstream region of the cognate target gene known as the promoter region, which triggers the enzyme, RNA polymerase, to transcribe DNA to RNA. TF can therefore be viewed as a class of specialized proteins that govern the on-off switch of gene expression through either repressing (down-regulation) or inducing (up-regulation) of output. The molecular readouts of a gene from two steps of gene expression are: (1) messenger RNA (mRNA), which is transcribed from DNA, and (2) protein, which is translated from RNA.

A gene regulatory network (GRN) is a network representing the orchestrated regulation of gene expression in a cell, where genes are nodes in this network and edges are regulatory relationships between genes. A typical GRN would consist of input signals or signaling pathways, regulatory proteins that receive and pass the signals, target genes, and the RNA and protein products (Figure 1). Because nuances in GRN output alteration could result in significant changes in cellular structure, physical capacity, or behavior of the cell, uncovering GRNs has become an essential task for a better understanding of cellular systems.

The problem of uncovering GRNs falls within the framework of system identification and is a traditional inverse problem. A difficulty unique to uncovering GRNs is the enormous scale of the problem involving hundreds or even thousands of genes, not to mention the nonlinearity and dynamics of regulation, inherent experimental errors, noisy readouts of expression levels,

and many unobserved factors. Obviously, the task calls for proper mathematical models and powerful inference algorithms.

B. Relevant Biological Data

The rationale for inferring GRN is rooted in the theory that drives systems biology. This emerging discipline postulates that, through careful data integration, the emergent properties of a modeled organism can be detected and insights gained that cannot be inferred from any single data set. A trademark of post-genomic systems biology research is the ever-increasing access to various biological data. A diverse collection of high-throughput biological data sources is currently available for elucidating GRNs and we discuss briefly in the following section the disparate types of data relevant to this research.

The different data types are categorized into (microarray) expression type and explanatory type:

1. *Expression Data* at the transcriptional level: Expression data measure the response of genes as a result of gene regulation at RNA levels. The most popular high throughput technology for producing gene expression data is microarray technology [6]. Microarray assays produce not only measurements of expression intensity but also the differential expression ratios under various experimental conditions.
2. *Next-generation Sequencing Data*: Next-generation sequencing is an emerging high-throughput, massively parallel DNA sequencing technology that is perceived to revolutionize the future genomic research. The next generation sequencing can process millions of sequence reads in parallel at a time rather than 96 of the conventional capillary-based sequencing with much reduced cost, which is reflected by its promise of the “\$1000” genome [7]. The key principle behind this technology is sequencing-by-synthesis, where millions of short sequence reads (35–250 bp) from a biological organism are synthesized and mapped to the corresponding location in the reference genome. The hits and distributions of each read are then recorded and analyzed. Next-generation sequencing has also inspired a host of “Seq-based” technology that will redefine the landscape of functional genomics. Compared with hybridization-based microarray technology, these new seq-based platforms bypass the long-standing technical difficulties associated with using DNA probes, cross-hybridization, etc. and greatly improve the data quality, sensitivity, scalability, and cost of microarray based experiments. The most noticeable application is the study of transcriptional binding sequences using chromatin immunoprecipitation, or ChIP-seq [8]. Other applications such as RNA-Seq enables discovery of small RNAs, RNAs from very low abundance groups, genome-wide single nucleotide polymorphisms (SNPs). In the context of gene regulation, the seq-based technology will be able to assay, in the real sense of a whole genome, the regulatory inputs and outputs for different cell types under different experimental or genetically varied conditions. These functional genomic measurement will certainly facilitate the study of gene regulatory networks. However, the new technology also introduces new computational challenges. For instance, there is 15–20% of the reads in human genome have multiple, ambiguous location mapping in the human genome due to gene paralogy. More importantly, the sheer volume of data generated by seq-based technology call for new, effective computational approaches. Signal processing will play an indispensable role in this new research frontier.
3. *Explanatory Data*: Explanatory data provide direct or indirect evidence of regulatory relationships between genes:

- a. ChIP-chip (Protein-DNA) interaction data: Central to the GRNs is the high-affinity binding between TFs and their cognate targets. A target gene of a specific transcription factor usually possesses a consensus binding motif, which is often located in the upstream region of a gene. The over-represented common regulatory motifs among a set of genes are indicative of co-regulated pattern and provide noisy evidence of the existence of a regulatory relationship [9].
- b. Expression at the translational level: The translational status of gene products (proteins) can be established in a large scale through proteomics experiments using 2-D gels and mass spectrometry. Because proteomic data reveal the existence of co-expressed or co-regulated protein complexes at specific developmental stages or/and specific cellular locations or compartments, it can help to delineate previous unrecognized architecture of signaling pathways within GRNs [10].
- c. Protein-protein interaction data: GRNs involve the fine regulation of proteins that work in a coordinated mode. The state-of-the-art yeast two-hybrid screening technology enables the characterization of thousands of proteins that are physically bound. Positive binding results can pinpoint yet unrecognized protein-protein interactions within GRNs [11].
- d. Gene deletion data: Systematic gene deletion produces measurements of phenotypic mortification due to gene deletion/mutation. Such measurements reveal direct regulatory relationship between one or a pair of deleted genes and their targets. Deletion data can be obtained by single-gene or double-gene mutation/deletion. However, due to the robustness of gene networks to resist attack to a single gene, single gene deletion data provide little discernable evidence for gene interaction, providing limited information on gene regulatory topology. In contrast, phenotype alterations are greatly enhanced by deletion of two genes, thus making assays of two gene deletion much more informative for elucidating gene interaction. *Synthetic genetic array (SGA)* is an assay technology for producing double-gene deletion data. Simply speaking, SGA produces an output array of double mutants by combining an input array of single mutants through mating and meiotic recombination [12].
- e. Other data: The cell signaling and dynamics in GRNs can also be studied by genetic and biochemical assays. For example, a direct means to study the causal relationship between genes or gene products is to knock out a specific gene to capture changes in system functionality. Recently, high throughput RNA interference (RNAi) technology makes it possible for a large-scale functional analysis of cellular traits when gene(s) are knocked out [13]. The structure of GRNs can also be consolidated with the characterized pathway information available at network databases such as KEGG database [14].

III. A Graphical Modeling Framework for Gene Regulation

The aim of this paper is to present a survey on the statistical models and the related inference algorithms. To make the survey more systematic and put a tangible connection between different existing models, we present a graphical modeling framework (GMF) for gene regulation. Developed in the field of artificial intelligence and signal processing, graphical modeling is the marriage between graph theory and probabilistic modeling [15]. It has several distinct features that suffice the modeling needs of this research. First, graphical modeling provides principles for the effective modeling of complex probabilistic systems such as those

at a cellular level and is very useful in sorting out the dependency between system variables and in visualizing otherwise obscure relationships. Secondly, graphical modeling allows the construction of complicated global models from simple local models and ensures that the model is extensible, allowing it to account for additional aspects of the system or new data sets. The general structure of graphical models permits relationships that are not allowed by Bayesian networks such as a circular regulatory relationships. Also, graphical models permit the use of continuous functions in the gene networks, potentially resulting in more accurate models than Bayesian networks or Boolean networks. In summary, graphical models are more appropriate to the modeling at a systems level and flexible enough to adapt to changes in knowledge. As a result, graphical models are better suited to take on the modeling needs of GRNs research. In addition, graphical modeling itself is general enough to include many existing models such as hidden Markov models and Markov Random Fields [15] as special cases. The framework thus provides the scaffolding on which the review on different models and inference algorithms can be systematically assembled.

The proposed GMF for two samples is depicted in Figure 2. The model has a layered structure with a total of 5 layers: explanatory data (including sequence data, protein interaction data, etc.), model parameters, unidentified factors, gene expressions, and microarray data. Specifically, the unidentified factors are elements that are potentially involved in gene regulation but whose identities are not explicitly defined due to lack of data or information. Gene expression denotes the molecular response of gene regulation at the RNA level. The microarray data represents the noisy measurements of gene expression; in each layer, a box is used to represent the associated *variable set* measured at a particular sample slice. Depending on the meaning of variable sets and sample index t , the GMF can be used to model either time series gene regulation such as cell cycles or static experiments with independent samples:

- *Modeling time series experiments:* In this case, t represents the specific sampling time instance. There are actually multiple slices and the two depicted here are only representative. Particularly, $\mathbf{y}_t$ denotes a set of microarray samples of all target genes measured at sampling time t , and the model parameter set includes the connectivity or topology variable S_t and model parameters θ_t .
- *Modeling static experiments:* In this case, t can take only two values, say 1 and 2, and has no specific meanings. Slice 2 is simply a copy of slice 1, i.e., $\mathbf{y}_1 = \mathbf{y}_2$, $S_1 = S_2$, etc. The variables in each set are assumed independent. The correlation between the variables is only modeled across slices. There is, in fact, only one slice of variable sets and the additional slice is added to facilitate modeling the relationship between regulators and elements under regulation.

The shaded boxes in the GMF (Microarray Data and Explanatory Data) indicate that the variable sets are observed. The arrows imply the direct association between the connected sets and the direction indicates a general meaning of ‘conditioning’ in the probability distributions. For example, we say $\mathbf{x}_{t+1}$, the gene expression levels at $t + 1$, depend on the expression levels $\mathbf{x}_t$, the unidentified factors $\mathbf{h}_t$, the connectivity S_{t+1} , and model parameters θ_{t+1} . This relationship is further quantified by the conditional distribution $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{h}_t, S_{t+1}, \theta_{t+1})$. The horizontal arrows would denote the dynamics in cell cycles if we model a time series experiment. In much of this review, we assume that the topology of the network remains unchanged, and thus the time index is dropped from S , θ , $\mathbf{h}$ in what follows, unless otherwise stated.

Under the framework, an existing model in the literature can be considered as a sub-model in the framework, consisting of a combination of different layers. This model can be then effectively broken down according to the layers of the framework. This process consists of two steps:

1. Identify the layers that the surveyed model addresses;
2. Translate the surveyed model into the conditional distributions of the parent sets at time $t + 1$.

After decomposing each surveyed model into layers, we can assemble different existing models by layers into the GMF. The subsequent survey of the models will be naturally conducted based on the layers.

IV. Survey of different models for gene regulation

We survey in this section different models proposed in the literature based on the GMF. As indicated before, surveys will be conducted based on layers with the order going from the microarray data layer to the explanatory data layer.

A. Modeling of Microarray Data

We review in this section the modeling of microarray data. In the GMF, the modeling is equivalent to defining the distribution $p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1})$.

1) Fully correlated model—In most of the current work [16], [17], [18], it is assumed that $p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) = \delta(\mathbf{y}_{t+1} - \mathbf{x}_{t+1})$ or $\mathbf{y}_{t+1} = \mathbf{x}_{t+1}$. This model is based on the assumption that microarray data is fully correlated with gene expression. It is used mainly for ease of inference. However, this assumption is oversimplified and this model does not coincide well with the real microarray experiments, which are fundamentally stochastic and noisy. Nonetheless, it has been widely adopted especially when emphasis is given to modeling gene interactions in the gene expression layer, in which case simple model is preferred to tradeoff computational complexity.

2) Linear Gaussian model—To capture the stochastic phenomena of microarray experiments, in [19], a linear stochastic model is applied and expressed as

$$\mathbf{y}_{t+1} = \mathbf{A}\mathbf{x}_{t+1} + \mathbf{a}_{obs} + \mathbf{v}_{t+1} \quad (1)$$

where $\mathbf{A}$ is a projection matrix and is assumed to be an identity matrix, $\mathbf{a}_{obs}$ is an observation adjustment vector, and $\mathbf{v}_{t+1}$ is white Gaussian noise vector with the covariance matrix $\sigma_{obs}^2 \mathbf{I}$. It is not difficult to see that the conditional distribution $p(\mathbf{y}_t|\mathbf{x}_t)$ is expressed by

$$p(\mathbf{y}_t|\mathbf{x}_t) = \mathcal{N}(\mathbf{A}\mathbf{x}_t + \mathbf{a}_{obs}, \sigma_{obs}^2 \mathbf{I}). \quad (2)$$

This model is a more accurate but at same time more complicate model than the fully correlated model. However, the assumption concerning white Gaussian noise may still not be realistic, and it is argued in [20] that the microarray data follow a skewed distribution. Nevertheless, the white Gaussian noise assumption faceplates the learning of networks, thus striking a balance between model accuracy and inference complexity.

3) Gaussian model with discrete expression levels—A factor graph representation is applied in [21] to model the relationship between the microarray data and gene expression. In that paper, the authors consider discrete expression levels, i.e., each variable in vector $\mathbf{x}_t$ is discrete. The continuous microarray data $\mathbf{y}_t$ are therefore noisy observations of the discrete expression levels. The relationship between the discrete expression levels and the continuous microarray data is described by a stochastic discretization function $\psi^d(\mathbf{x}_{i,t+1}, \mathbf{y}_{i,t+1})$, where the

index i is used to indicate the i th variable of the vector. Through this discretization function the conditional distribution can be determined. In [21], a mixture of Gaussian discretization function is applied and the corresponding conditional distribution is

$$p(y_{i,t+1}|\mathbf{x}_{i,t+1})=\mathcal{N}(\mu_i, \sigma_i) \quad (3)$$

where μ_i and σ_i depend on $x_{i,t+1}$ and the experimental evidence can be learned from the data directly. Compared with *linear Gaussian model*, it is still based on Gaussian noise assumption but it could potentially better due to the underlying nonlinear model for expression levels.

B. Modeling the Gene Expression Layer

The objective here is to model the dynamics in gene regulation. Under this framework, it concerns defining the conditional distributions of node $\mathbf{x}_{t+1}$ or $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{h}_t, \theta_{t+1}, S_{t+1})$.

Various linear and nonlinear models for continuous and discrete gene expression levels have been studied for gene regulation. These models include linear Gaussian models, the decision tree based mixture Gaussian model [22], the Sigmoidal Squashing function, spline interpolation [23], etc. Each one of them is reviewed briefly.

1) Dynamic Bayesian Networks - Multinomial Distribution—When the expression levels are considered discrete, a popular approach as in [24], [25], [18], [17] is to model the gene regulatory relationships by the discrete Bayesian networks (DBNs). The DBNs possess the potential of probabilistic models while allowing for nonlinear relationships between genes and the natural addition of prior information in the form of a soft constraint. Nonlinearity is captured by assigning a multinomial distribution to the conditional distributions of the regulatory dynamics. Equivalently, the DBN assumes a first order time homogenous Markov chain for gene regulation. Under the GMF, this modeling leads to the following conditional distribution

$$p(\mathbf{x}_{t+1,i}|\mathbf{x}_t, \theta, S)=\xi_{i,S}^l \quad (4)$$

where $\mathbf{x}_{t+1,i}$ represents the expression level of gene i , l denotes the l -th level, and $\xi_{i,S}^l$ is the conditional probability of gene i taking level l given the parents at t and the topology S . The unknown model parameters θ then consist of all the conditional probabilities. One reason for the popularity of this model is that with the Dirichlet prior, it will lead to a tractable marginal likelihood, thus making the network inference relatively easier. However, the drawback of this model is in discretization, which incurs loss of information.

2) Factor Graph Models—One major problem with BNs is its inability to model loopy relationships. Factor graphs (FGs) overcome this problem as they can decompose interactions between variables more flexibly than Bayesian networks (BNs) or Markov random fields (MRFs). Consequently, FGs are more useful for describing models that involve a large number of overlapping relationships between variables like genetic regulatory networks. FGs have been applied to the problem of inferring regulatory networks in [26], [27]. A Factor Graph is a bipartite undirected graph that expresses how a global function of several variables factors into a product of local functions. FGs have two types of nodes (variable nodes and factor nodes), where the variable nodes represent the variables in the model whereas the factor nodes represent each local function. Only edges between nodes of different classes are allowed and a variable

node is connected to a function node if and only if this variable is an argument of the corresponding local function. Like BNs or MRFs, FGs model the probability functions, which can be factorized into a product of local functions. FGs will essentially be equivalent DBNs when used to model dynamics of gene expression but it is more powerful to also model loopy interaction of gene expressions at each time sample. As a result, the conditional distribution is expressed as

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \theta, S) = \prod_i^G p(x_{t+1,i}|\mathbf{pa}_i, \theta_i, S) \quad (5)$$

where $\mathbf{pa}_i$ is the expression levels of a set of *parents* of gene i . Parents and factorization in (5) are defined by the graph. Moreover, any variable of either $\mathbf{x}_{t+1}$ or $\mathbf{x}_t$ except $x_{t+1,i}$ can be a potential parent, which essentially allows loopy relationship between variables. $p(x_{t+1,i}|\mathbf{pa}_i, \theta_i, S)$ can assume different probabilistic models as appropriate. In [26], [27], discrete expression levels were assumed and a multinomial model was adopted.

3) Probabilistic Boolean Networks—An alternative to the Multinomial regulatory model are probabilistic Boolean networks (PBNs) [2]. In PBNs, each gene is assumed to exhibit only binary expression levels with 1 representing gene expressed and 0 as not expressed. The dynamic regulatory relationship is modeled in a PBN by a list of Boolean function sets and the i th set $F_i = (f_1^{(i)}, \dots, f_{\zeta(i)}^{(i)})$ contains $\zeta(i)$ possible Boolean functions, or predictors, used to predict the state of gene i . At any given time, only one predictor, say $f_k^{(i)}$, is chosen to determine the state of gene i and the probability of choosing the predictor is represented by $c_k^{(i)}$. Since the inputs of $f_k^{(i)}$ are the states of the regulating genes, the topology of the GRNs at any time is defined through the predictors chosen. The topology will be fixed given the predictors for each gene, but nonetheless different predictors of each gene can result in the same GRN topology. Suppose that, among the $\zeta(i)$ possible predictors of gene i , K_i will lead to the same GRN topology and we collect indices of these K_i predictors in an integer set $\mathcal{K}_i$, then the desired conditional probability can be expressed as follows

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \theta, S) = \prod_{i=1}^G \sum_{k \in \mathcal{K}_i} p(x_{i,t+1}|\mathbf{x}_t, \theta_{i,t+1}, f_k^{(i)}) c_k^{(i)} \quad (6)$$

where G is the total number of genes. The detailed expression of $p(x_{i,t+1}|\mathbf{x}_t, \theta_{i,t+1}, f_k^{(i)})$ in (6) depends on the specific function form of the predictor $f_k^{(i)}$. For deterministic Boolean function as used in [2], we have

$$p(x_{i,t+1}|\mathbf{x}_t, \theta_{i,t+1}, f_k^{(i)}) = 1 - |f_k^{(i)}(\mathbf{x}_t) - x_{i,t+1}| \in \{0, 1\}. \quad (7)$$

In [28], a probabilistic neural network was used for the predictor $f_k^{(i)}$. The corresponding conditional probability when the Gaussian basis function is used can be written as

$$p(x_{i,t+1}|\mathbf{x}_t, \sigma_i, f_k^{(i)}) = (2\pi\sigma_i)^{-N/2} \exp\left(-\frac{1}{2\sigma_i} \|x_{i,t+1} - f_k^{(i)}(\mathbf{x}_t)\|^2\right) \quad (8)$$

where σ_i is the noise variance and N is the number of observations.

As we can see from (6), PBNs, like dBNs, essentially model the gene regulation with Markov chains. Therefore, as pointed out in [29], PBNs will be the same as dBNs in the sense they both can provide the same network model with the same conditional probabilities. However, PBNs are also fundamentally different from dBNs because it is constructed by modeling the on-off behavior of gene regulation

4) Boolean functions with random inputs—In [30], the gene regulatory relationships are modeled using Boolean logic functions with probabilistic inputs. For this model, the expression levels $\mathbf{x}_t$ are considered to be discrete (either 0 or 1), and it is also assumed that the influence of each regulator to the element regulated is independent of other possible regulators. These models were called “noisy OR-Gate” and “noisy AND-Gate”. The regulatory process is not deterministic since each gene $x_{j,t}$ can regulate the gene $x_{i,t+1}$ with probability ρ_{ij} and can fail to do this with probability $1 - \rho_{ij}$.

If the model with “OR”-activation is considered, the conditional probability distribution can be written as

$$p(x_{i,t+1}|\mathbf{x}_t, \theta, S) = \delta_{x_{i,t+1},0} \prod_{j=1}^n (1 - \rho_{ij})^{x_{j,t}} + \delta_{x_{i,t+1},1} \left(1 - \prod_{j=1}^n (1 - \rho_{ij})^{x_{j,t}} \right) \quad (9)$$

where $\delta_{i,j}$ is the Kronecker Delta and it is equal to one, if $i = j$ and zero otherwise.

This Boolean function with random input is similar to the PBNs in the sense that they both model regulatory processes using Boolean logic semantics. However, in PBNs, the randomness is introduced into the model through randomly selecting one possible Boolean function for each node in the networks while each Boolean function itself is deterministic. In Boolean logic function based models, the source of uncertainty in genetic regulation is modeled through the independent random inputs of Boolean functions. From a modeling perspective, PBNs intend to model stochastic regulation yet Boolean logic functions targets randomness in gene expression.

5) Graphical Gaussian Models—Graphical Gaussian Models (GGMs) have recently become a popular tool to study gene networks [31], [32], [33], [34]. GGM emulates the continuous expression levels directly. The key idea behind GGMs is to use partial correlations as a measure of the conditional independence between any two genes in a network. This is achieved by measuring the correlation between two genes after the effects of all other genes have been removed. By considering the partial correlation instead of simply the correlation, the GGM is able to distinguish between direct interactions, indirect interactions and regulations by a common gene. Moreover, although they behave similarly to the Bayesian networks, GGMs possess the advantage of containing only undirected edges, so they can be applied to networks with feedback loops. The statistical model for the relationship among the genes is represented as a graph, called the independence graph, whose nodes represent the variable under consideration and the edges represent direct interactions between genes.

In order to apply GGMs, the distribution $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \theta, S)$ is assumed to be a multivariate normal distribution with a mean equal to $\boldsymbol{\mu}$ and a positive definite covariance matrix $\boldsymbol{\Sigma}$. Thus, if we divide $\mathbf{x}_{t+1}$ into M clusters, the conditional distribution of cluster m can be written as a Gaussian distribution

$$p(\mathbf{x}_{t+1}^m|\mathbf{x}_t, \theta, S) = \mathcal{N}(\mu_m, \sigma_m^2) \quad (10)$$

The basic procedure in the classical GGM theory is the following: the partial correlation can be computed using the elements of the inverse variance matrix since it is known that conditional independence constraints are equivalent to specifying zeros in the inverse variance. So, once the inverse variance and the partial correlations are calculated, the independence graph is drawn. The rules state that an edge is not included in the graph if the absolute value of its partial correlation coefficient is less than a predefined threshold.

One of the major drawbacks of the GGMs is related to the difficulties derived from its application to high-dimensional data that goes from ill-conditioning problems to invalidity of the statistical tests used for selecting appropriate GGMs. One of the most extended solutions is to reduce the dimension of the data by clustering. Another possible solution is the one based on the calculation of the limited order partial correlations [33]. The main disadvantage of this approach is the fact that it remains unclear whether the missing edges indicate conditional or marginal independence.

6) Linear Gaussian model—In the case of continuous gene expression levels, linear Gaussian models have been widely used in modeling the dynamics of gene regulation [16], [19], [35]. In this model, the expression level of each gene is assumed to be the result of a linear combination of the expression levels of the regulating genes at a previous sample time. Mathematically, it can be expressed as

$$\mathbf{x}_{t+1} = \mathbf{W}\mathbf{x}_t + \tilde{\mathbf{B}}\mathbf{h}_t + \mathbf{u}_t \quad (11)$$

where $\mathbf{W}$ is the regulation weight matrix independent of time t , $\tilde{\mathbf{B}}$ is another weight matrix that describes the influence of unidentified factors to gene expression, and $\mathbf{u}$ is assumed to be white Gaussian noise with covariance matrix $\sigma_i^2 \mathbf{I}$. The topology S determines the sparsity of the matrix, i.e., the i, j th element of $\mathbf{W}$ will be zero if gene j is not a regulator of gene i . The nonzero element of the weight matrix is indicative of the degree and the types of the regulation [16], [35]. A gene is up-regulated if the weight is positive and is down-regulated otherwise. The magnitude (absolute value) of the weight indicates the degree of regulation. From (11), we obtain that the conditional distribution is a Gaussian distribution, i.e.

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{h}_t, \theta, S) = \mathcal{N}(\mathbf{W}_s \mathbf{x}_t + \tilde{\mathbf{B}} \mathbf{h}_t, \sigma_i^2 \mathbf{I}). \quad (12)$$

where $\theta_s = \{\mathbf{W}_s, \tilde{\mathbf{B}}, \sigma_i^2\}$. Linear Gaussian models can be considered as a counterpart of the multinomial regulatory model under discrete expression levels. A similar computational advantage is possessed by linear Gaussian models as well. When the conjugate Gaussian-Inverse-Gamma distribution is applied for model parameters, the marginal likelihood needed for topology inference can be obtained analytically. Compared with multinomial regulatory

model, linear Gaussian model has the advantage of modeling the continues expression level directly but it cannot capture the nonlinearity of gene regulation.

7) Gaussian model with Sigmoidal Squash function—Despite the popularity and computational advantage of linear Gaussian models, in real world gene regulatory networks, the dependencies are known to be non-linear (for example, a saturation effect is expected). In this model, the expression level of each gene at a given sample time can be written as

$$\mathbf{x}_{t+1} = g(\mathbf{W}_s \mathbf{x}_t + \tilde{\mathbf{B}} \mathbf{h}_t) + \mathbf{u}_t \quad (13)$$

where the function $g(\cdot)$ is nonlinear. An example of $g(\cdot)$ is the Sigmoidal Squash function. In these cases, the conditional distribution is still Gaussian

$$p(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{h}_t, \theta_s, S) = \mathcal{N}(g(\mathbf{W}_s \mathbf{x}_t + \tilde{\mathbf{B}} \mathbf{h}_t), \sigma_i^2 \mathbf{I}). \quad (14)$$

where $\theta_s = \{\mathbf{W}, \tilde{\mathbf{B}}, \sigma_i^2\}$. However, now the mean of the density is expressed as a non-linear function of the parents. The disadvantage of this model is that the nonlinearity increases the complexity of the network inference.

8) Gaussian model with Spline interpolation—In addition to Sigmoidal function for capturing the nonlinear regulatory relationship, in [23] a nonparametric additive regression model with Gaussian noise is assumed. Again, the expression levels $\mathbf{x}_t$ are considered to be continuous variables and a non-parametric regression model based on B-splines is used

$$x_{i,t+1} = \bar{m}(\mathbf{x}_t) + u_i \quad (15)$$

where u_i is additive Gaussian noise with zero mean and variance σ_i^2 . And

$$\bar{m}(\mathbf{x}_t) = m_{i,1}(x_{1,t}) + \dots + m_{i,P_i}(x_{P_i,t}) \quad (16)$$

where P_i is the number of parents of the i -th gene and $m_{i,j}(x_{j,t})$ is a smooth function from $\mathcal{R}$ to $\mathcal{R}$ and can be expressed by using the linear combination of basis functions

$$m_{i,j}(x_{j,t}) = \sum_{r=1}^{R_{rj}} \gamma_{rj}^{(i)} b_{rj}^{(i)}(x_{j,t}), \quad j=1, \dots, P_i \quad (17)$$

where $\gamma^{(i)} = [\gamma_{1j}^{(i)}, \dots, \gamma_{R_{rj}j}^{(i)}]$ are coefficient parameters and $\{b_{1j}^{(i)}(\cdot), \dots, b_{R_{rj}j}^{(i)}(\cdot)\}$ is the prescribed set of basis functions. Taking all the previous expressions into account, the conditional probability can be written as:

$$p(x_{i,t+1}|\mathbf{x}_t, \theta, S) = (2\pi\sigma_i^2)^{-1/2} \exp\left(-\frac{1}{2\sigma_i^2}|x_{i,t+1} - \mu(\mathbf{x}_t)|^2\right) \quad (18)$$

where $\theta = \{\sigma_i^2, \gamma^{(i)}\}$. Compared with Sigmoidal Squash function, Spline interpolation could be more flexible in modeling different regulatory relationship and it is computationally simpler.

9) Gaussian Kernel functions—In [36], a pair-wise interaction was considered. To model the conditional distribution $p(x_{i,t+1}|x_{j,t}, \theta, S) \forall i, j$, a Kernel estimator was introduced. In detail, given a set of gene expression measurements the joint distribution $p(x_{i,t+1}|x_{j,t}, \theta)$ is first modeled by

$$p(x_{i,t+1}, x_{j,t}|\theta, S) = \frac{1}{M_e} \sum_i^{M_e} \varsigma^{-2} \Upsilon(\varsigma^{-1}|\mathbf{z} - \mathbf{z}^{(m_e)}|) \quad (19)$$

where $\mathbf{z} = [x_{i,t+1}, x_{j,t}]^T$, $\mathbf{z}^{(m_e)}$ for $m_e = 1, \dots, M_e$ are M_e measurements of $\mathbf{z}$, $\Upsilon(\cdot)$ is the 2-D Gaussian Kernel defined by

$$\Upsilon(\mathbf{z}) = 2\pi e^{-\frac{|\mathbf{z}|^2}{2}} \quad (20)$$

and ς controls the width of the kernel. For the time series stationary Markov assumption, $\mathbf{z}^{(m_e)}$ can be the measurement pair at consecutive sampling times. Using the joint distribution and its marginals, the conditional distribution can be obtained by

$$p(x_{i,t+1}|x_{j,t}, \theta, S) = \frac{p(x_{i,t+1}, x_{j,t}|\theta)}{p(x_{j,t}|\theta, S)} \quad (21)$$

where $p(x_{j,t}|\theta, S)$ is the marginal distribution of $p(x_{i,t+1}, x_{j,t}|\theta, S)$. The kernel function model is a nonparametric model. It is desirable due to its minimal effect from outliers in experimental data. Being a class of nonparametric models, Gaussian Kernel functions can be flexible and robust in modeling different regulatory relationships. However, since modeling complexity and error increase drastically with the dimension of x , kernel functions are limited to model pair-wise interactions.

10) Regression tree with Gaussian distribution—Another approach to model the nonlinear regulation with continuous expression levels is to use a regression tree. In [22], the regression tree is applied to represent the regulatory relationships between *Modules*, though the same can be used for genes as well. In a module network, each module includes a set of genes with the same statistical behavior, the same set of parents, and the same logical probabilistic model. The conditional distribution of the module networks is then modeled with a regression tree. Each tree has nodes and leaves, where a node represents a condition of the parent gene expression, for example, “Is $\mathbf{x}_{i,t+1} > 2.1$?”. Descending from each node there are two child nodes, each coming as an outcome of the condition of the parent node. The choice on the children depends on whether the condition is true or not. Leaves are the last layer of the regression tree and each leaf is associated with a (conditional) distribution of the module. The

final conditional distribution depends on the last leaf reached in the tree. In [22], each leaf has a univariate Gaussian distribution associated to represent the conditional distribution of the module. Thus, if we divide $\mathbf{x}_{t+1}$ into K modules, the conditional distributions of module k can be written as a Gaussian distribution

$$p(M_{k,t+1}|\mathbf{x}_t, \theta, S) = \mathcal{N}(\mu_\eta, \sigma_\eta^2) \quad (22)$$

where $M_{k,t+1}$ represents the expression of genes in the k th module, η is an indicator function of $\mathbf{x}_t$, which is equal to j , only if the values of vector $\mathbf{x}_t$ lead towards the j -th leaf, and θ contains the set of parameters of the Gaussian distribution. Note that the nonlinear regulation is included through the indicator η . The conditional distribution $p(M_{k,t+1}|\mathbf{x}_t)$ is a mixture Gaussian distribution. Since inference of mixture Gaussian models has been studied extensively, computational complexity of this model, though higher than linear Gaussian models, is considered manageable. However, determining the number of modules and decision rules can be difficult.

11) Linear Gaussian Model with Non-uniform Sampling Interval—In microarray experiments, data samples are obtained with unequal (time) intervals. The modeling of this issue is considered in [37], where it starts with the simplest possible differential equations for regulatory system model;

$$\frac{d\mathbf{x}_t}{dt} = \mathbf{M}\mathbf{x}_0 \quad (23)$$

where $\mathbf{M}$ is the model coefficient. The solution to this system is:

$$\mathbf{x}_t = e^{\mathbf{M}t} \mathbf{x}_0 \quad (24)$$

Since equation (24) is nonlinear in $\mathbf{M}$, it will be difficult to estimate $\mathbf{M}$ using experimental data in the network inference step. Thus, the differential equation (23) is approximated by the following difference equation:

$$\mathbf{x}_{t_j} - \mathbf{x}_{t_{j-1}} = (t_j - t_{j-1})\mathbf{M}\mathbf{x}_{t_{j-1}} + n_{t_j} \quad (25)$$

where $t_j, j \in \{1, \dots, n\}$ are the sampling (time) indices taken and n_{t_j} is Gaussian distributed and accounts for the error. This model includes information about the sampling interval and it is useful for modeling unevenly sampled data. Taking all the previous expressions into account, the conditional probability can be written as:

$$p(\mathbf{x}_{t_j}|\mathbf{x}_{t_{j-1}}, \theta, S) = \mathcal{N}(\mathbf{x}_{t_j} - (1 + (t_j - t_{j-1})\mathbf{M})\mathbf{x}_{t_{j-1}}, \widehat{\sigma}^2) \quad (26)$$

where θ includes $\mathbf{M}$ and σ^2 . The approximation of (25) becomes less accurate with the increase of the sampling interval $t_j - t_{j-1}$.

C. Modeling the Unidentified Factors Layer

The objective is to model the dynamics of unidentified factors in gene regulation. Under the framework, this is equivalent to defining the distribution $p(\mathbf{h}_{t+1}|\mathbf{x}_t, \mathbf{h}_t, \boldsymbol{\theta}, S)$. Unidentified factors are any elements that are important to gene regulation but not directly observed. Within the broader context of a cell, gene regulation interacts with metabolism, signal transduction, replication, recombination and repair, and a variety of other processes. However, in most of the existing models, only mRNA levels are included and other elements are left out for either computational simplicity or the assumption that, e.g. protein expression levels are likely to be reflected by the corresponding mRNA levels. These assumptions are not always true. For example, when modeling gene expression data on a process involving metabolism, an effort should be made to model the perturbed GRNs given the changes of flux in essential metabolites and nutrient supplies. Also, the available data might not always provide evidence concerning all the regulators of the target gene and ignoring the existence of unobserved regulators in modeling, though lowering the computational requirement will certainly change the topology of the network. We call any elements that are important to gene regulation but not directly observed unidentified variables. Including them in modeling gene regulation will present a more accurate picture.

In [16], a linear model is applied to model dynamics of the unidentified variables as well as their interaction with the gene expression

$$\mathbf{h}_{t+1} = \mathbf{C}\mathbf{h}_t + \mathbf{D}\mathbf{x}_t + \mathbf{v}_t \quad (27)$$

where the matrix $\mathbf{C}$ is the dynamics matrix of $\mathbf{h}_t$, the matrix $\mathbf{D}$ models the influence of gene expression values at t on the value of unidentified states at $t + 1$, and $\mathbf{v}_t$ is Gaussian noise with variance σ_h^2 . As a result, the conditional distribution $p(\mathbf{h}_{t+1}|\mathbf{h}_t, \mathbf{x}_t, \boldsymbol{\theta}, S)$ is also Gaussian and can be expressed as

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{h}_t, \boldsymbol{\theta}, S) = \mathcal{N}(\mathbf{C}\mathbf{h}_t + \mathbf{D}\mathbf{x}_t, \sigma_h^2 \mathbf{I}) \quad (28)$$

where $\mathbf{C}$, $\mathbf{D}$, and σ_h^2 are model parameters.

Although adding a hidden layer in the overall modeling is more accurate in modeling, nevertheless it drastically increases the computational complexity, making the use of more complicated nonlinear model infeasible.

D. Modeling the Parameter Layers

Assuming independence between S and $\boldsymbol{\theta}$, the task here concerns defining the distributions $p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t, S_{t+1})$, $p(S_{t+1}|S_t)$, and $p(S_{t+1}|\mathbf{d}_{t+1})$. $p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t, S)$ and $p(S_{t+1}|S_t)$ model the time (experiment) variation of the respective model parameters and network topology. $p(S|\mathbf{d}_{t+1})$ provides the evidence obtained from the explanatory data of the network topology and it has been referred to as the prior distribution in the literature. We discuss in the following the specific expressions of these three distributions.

1) Modeling the variation of the model coefficients—Most of the existing work assumes static or time invariant model coefficients, i.e., $p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t, S_{t+1}) = \delta(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t)$. Very little attention has been paid to time varying parameters even though the parameters such as the weight of regulation could change over time. Efforts are mainly hampered by the increased

complexity in addition to an already intricate inference task. Yet, a model similar to that for target movement in target tracking was proposed in [38] for the time varying regulatory weights of a linear-Gaussian expression model (11) as

$$\begin{pmatrix} w_{ij}(t+1) \\ \dot{w}_{ij}(t+1) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} w_{ij}(t) \\ \dot{w}_{ij}(t) \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} v_{ij}(t+1) \quad (29)$$

where w_{ij} is the ij th element of the weight matrix $\mathbf{W}$, $\dot{w}_{ij}$ is the first order derivative of w_{ij} that describes the rate of the weight variation, and $v_{ij}(t+1)$ is white Gaussian noise. As a result, the distributions $p(\theta_{t+1}|\theta_t, S_{t+1})$ follows a Gaussian distribution. Like the hidden level, modeling of variation of model coefficient is hampered by the increased complexity that it introduces.

2) Modeling the variation of the network topology—It is common and arguably reasonable to consider a stationary gene network, where the topology of the network does not change between t and $t+1$. The assumption thus implies $p(S_{t+1}|S_t) = \delta(S_{t+1}-S_t)$. Yet, non-stationary network can be also modeled by specifying a proper distribution for the conditional distribution of the topology. However, such practice can be computationally infeasible.

E. The prior distribution of the topology

In the GMF, we are concerned with the conditional distribution $p(S|\mathbf{d})$, or the prior distribution of the topology. Note again, that we ignore the subscript t , assuming an underlying stationary process. This prior distribution reflects our prior belief about the network connectivity before observing the microarray data. Through this prior distribution, integration of other types of data and microarray data is possible. Depending on the existence of explanatory data $\mathbf{d}$, different types of the prior distribution should be assumed.

1) The prior distribution without explanatory data—Since $\mathbf{d}$ is unavailable, the prior distribution is simplified as $p(S)$. In theory, when no prior evidence is available, the prior distribution is constructed to be as *noninformative* as possible. In practice, the prior distributions are often designed to impose regularity constraints and to exploit the local structure of data models.

To construct the prior distribution $p(S)$, it is assumed that the network is decomposable, i.e.,

$$p(S) = \prod_{i=1}^G p(S_i) \quad (30)$$

where S_i is the subnetwork that defines the regulation towards gene i . The attention can be then focused on the design of the prior distribution of a subnetwork.

The Power law distribution: The power law distribution is investigated in [24], which is defined as

$$p(S_i) = a P_i^{-b} \left(\frac{G}{P_i} \right) \quad (31)$$

where a and b are the coefficients of the distribution and P_i represents the total number of regulators or parents of gene i . The power law distribution penalizes the network with larger connectivity and is adopted to describe the scale invariance property of the biological networks.

The DL criterion: A different choice is based on the description length (DL) criterion [39], and the prior distribution has the following expression

$$p(S_i) \propto \kappa^{-L(S_i)} \quad (32)$$

where $L(S_i)$ is a so-called description length (DL) of the network and κ is commonly chosen as either e or 2 depending on the regulatory models defined at the gene expression layer. The DL of the subnetwork S_i is defined as

$$L(S_i) = \log_{\kappa} G + \log_{\kappa} \binom{G}{P_i}. \quad (33)$$

Like the power law prior, the DL prior also encourages networks with the small number of conductivities.

The independent Bernoulli prior: The third choice is based on the *a priori* assumption that a gene has the probability q to be a regulator [40]. As a result, the prior distribution of S_i is defined as

$$p(S_i) = q^{P_i} (1 - q)^{G - P_i}. \quad (34)$$

Since, under the above prior, Gq denotes the mean number of the regulators, q can be chosen based on the prior knowledge of the average number of the regulators.

2) The prior distribution with explanatory data—Where explanatory data are available, the prior distribution $p(S|\mathbf{d})$ should be constructed differently for the different types of explanatory data. Since the evidence carried by explanatory data is often qualitative and deterministic binary information as “yes” or “no”. Even when it is probabilistic, the information is often in the form of confidence intervals or p -values. Care must be taken to map the evidence to the prior distributions.

The Gibbs random field: In [41], a Gibbs random field was proposed for the mapping and various data types were tested. The specific datatypes that have been studied for the mapping include protein-protein interaction, protein-DNA interaction, motif sequence information, and known networks and pathways in the existing database. To model the prior from these data, the following Gibbs distribution as used

$$p(S|\mathbf{d}_{t+1}) = Z^{-1} \exp - E(S) \quad (35)$$

where Z is the normalizing constant and $E(s)$ is the total energy of the network S and is calculated as

$$E(S) = \sum_{i=1}^G E_i = \sum_{i=1}^G \sum_{j \in \mathcal{L}_i} U_{ij} \quad (36)$$

where $\mathcal{L}_i$ is the index set of the parents of gene i and U_{ij} is the local energy between gene i and j . U_{ij} takes discrete values reflecting the state of the explanatory data about the interactions between gene i and j . As a common example, $U_{ij} \in \{H_1, H_2\}$ with $0 < H_1 < H_2$. As such, we assume H_1 for U_{ij} if, for instance, the protein-protein interaction data suggest that gene j regulates gene i , and H_2 otherwise. In using the Gibbs distribution to model the prior knowledge from explanatory data, it can be quite problematic to calculate Z and to determine the proper values for H_1 and H_2 . Practical strategies have been proposed in [41] for mapping deterministic prior evidence. Yet, no discussion has been provided on how to interpret the probabilistic evidence with the Gibbs distribution.

The independent Bernoulli prior for the p -value: An alternative to the Gibbs distribution is the independent Bernoulli prior that is constructed for each edge [42]. In [42], this model was applied to construct the prior from the data generated from the location analysis of yeast TFs [43]. p -values were produced from the analysis of the confidence of genes being TFs. The objective of mapping is to translate the p -values to prior probability of existence of edges. Specifically, it is assumed that

$$\begin{aligned} P(S|\mathbf{d}_{t+1}) &= \prod_{n=1}^{N_e} (\delta(E_n \in S)P(E_n \in S|\mathbf{d}) + \delta(E_n \notin S)P(E_n \notin S|\mathbf{d})) \\ &= \prod_{n=1}^{N_e} (\delta(E_n \in S)P(E_n \in S|p_n) + \delta(E_n \notin S)P(E_n \notin S|p_n)) \end{aligned} \quad (37)$$

where it is assumed that there are totally N_e edges in the topology space, E_n denotes the n th edge, $\delta(\cdot)$ is 1 when the argument is true and 0 otherwise, and p_n is the p -value for the n th edge. This last equality is due to the fact that evidence from data is presented by the p -value. Now, given the p -values, we need to define the distributions $p(E_i \in S|p_n)$, from which $p(E_i \notin S|p_n)$ can be easily obtained. According to the definition of p -value, it is assumed that p_n follows a uniform distribution between 0 and 1. Further, p_n is assumed in [44] to follow a truncate exponential distribution with parameter λ if $E_n \in S$. Then, according to the Bayes rule we can obtain

$$p(E_n \in S|p_n, \lambda) = \frac{\lambda e^{-\lambda p_n} \beta}{\lambda e^{-\lambda p_n} \beta + (1 - e^{-\lambda})(1 - \beta)} \quad (38)$$

where $\beta = p(E_n \in S)$ is the prior probability of E_i before observing the data $\mathbf{d}_{t+1}$. Assuming λ follows a uniform distribution between $[\lambda_1, \lambda_2]$, the desired prior distribution $p(E_n \in S|p_n)$ can be obtained by marginalizing λ in (38)

$$p(E_i \in S|p_n) = \int_a^b \frac{\lambda e^{-\lambda p_n} \beta}{\lambda e^{-\lambda p_n} \beta + (1 - e^{-\lambda})(1 - \beta)} d\lambda. \quad (39)$$

Even though the integration can not be solved analytically, numerical integration can be applied.

Markov random field: In [45], Markov random field (MRF) was proposed to model the binary protein interaction data. Unlike the Gibbs random field or the independent Bernoulli prior, which models only the pairwise edge directly, Markov random field is defined on a set of vertices $\mathcal{V} = \{V_1, \dots, V_2\}$ and each vertex is associated with a gene. The vertex variables can be used to represent different entity of interest of genes. In addition to direct interaction between two genes, they represent the cluster or pathway indices of each gene in [45]. In this context, genes in a cluster are considered to belong the same pathway. Given a set of binary protein interactions, the distribution of $\mathcal{V}$ can be modeled as

$$p(\mathcal{V}|\mathbf{d}_{t+1}) = Z^{-1} \prod_{g=1}^G \varphi_g(V_g) \prod_{E_{ij} \in \mathcal{E}} \varphi_{ij}(V_i, V_j) \quad (40)$$

where Z is the normalizing constant, φ_g is a potential function of V_g , E_{ij} represent the edge between gene i and j , $\mathcal{E}$ is the set of interacting edges defined by the protein interaction database, and φ_{ij} is a non-negative compatibility potential. This potential function specifies how “compatible” vertices i and j is. For example, if vertices define the association of clusters or pathways, the potential function can be defined as [46]

$$\varphi_{ij}(V_i, V_j) = \begin{cases} \alpha & V_i = V_j \\ 1 & \text{otherwise} \end{cases} \quad (41)$$

where $\alpha \geq 1$. α defines the degree of agreement or compatibility between pathway assignment and protein interaction. The larger α indication the two genes interacting at the protein level are more likely to be in the same pathway. Compared other models surveyed in the subsection, Markov random field model is more flexible. However, MRF cannot model directed relationship.

V. Survey of network inference algorithms

So far we have reviewed the existing models for each layer of the GMF. Once the models are chosen for describing the target gene regulatory systems, the next step is to infer the networks from data. The GMF at hand presents two clear objectives for inference tasks: learning the network topology and estimating the model parameters. In particular, the primary interest is to infer, based on microarray and explanatory data, the connectivity or topology S . The model parameters θ , which can provide information on regulatory intensity, cell cycle phase, etc., are of secondary interest. The unidentified factors $\mathbf{h}$ and gene expression $\mathbf{x}$ are also unknown but considered as nuisance parameters in Bayesian inference. Within the inference theory, topology learning is a model selection problem and parameter learning is an estimation problem. With these objectives, the inference algorithm can be systematically developed under the framework.

From a Bayesian perspective, parameter learning is naturally blended within the process of topology learning and the task is performed with respect to the *a posteriori* probabilities (APP) s of topology $p(S|\mathbf{y}, \mathbf{d})$. Depending on the learning objectives, the approaches can be classified into the point (hard) solutions and the probabilistic (soft) solutions. In what follows, we survey some of the salient algorithms for learning tasks under the two categories.

A. Point solutions - Generalized Bayesian score

The goal of point solutions is to determine the most likely topology supported by the data. It is an optimization problem in nature and the maximum *a posteriori* (MAP) [47] criterion is commonly adopted, i.e.

$$\begin{aligned}\hat{S}_{MAP} &= \arg \max_S p(S|\mathbf{y}, \mathbf{d}) \\ &= \arg \max_S GBScore(S)\end{aligned}\quad (42)$$

where $\hat{S}_{MAP}$ is the inferred topology under the MAP criterion and GBScore stands for the generalized Bayesian score and is defined as

$$\begin{aligned}GBScore(S) &= \log p(\mathbf{y}|S) + \log p(S|\mathbf{d}) \\ &= BScore(S) + \log p(S|\mathbf{d})\end{aligned}\quad (43)$$

where $BScore(S)$ is commonly referred ‘Bayesian score’ and $p(\mathbf{y}|S)$ is the marginalized likelihood calculated by

$$p(\mathbf{y}|S) = \int p(\mathbf{y}|\theta, \mathbf{h}, S) p(\theta, \mathbf{h}|S) d\theta d\mathbf{h}.\quad (44)$$

where $p(\theta, \mathbf{h}|S)$ is the prior distribution of the model parameters and unidentified variables. The Bayesian score $BScore(S)$ is a score function obtained solely from the marginal likelihood $p(\mathbf{y}|S)$. In using the Bayesian score for structure learning, the topology prior $p(S|\mathbf{d})$ is ignored either during the asymptotical analysis or under the noninformative assumption on the prior evidence of topology. Since the marginalized likelihood $p(\mathbf{y}|S)$ automatically penalizes complex topology through marginalization, the phenomenon known as Ockham’s razor, overfitting can be avoided with the Bayesian score alone. Yet, when prior evidence from the explanatory data is available, it is more appropriate to use the GB-Score for learning.

The specific point solution for learning based on the GB-Score can be classified depending on if the marginalized likelihood $p(\mathbf{y}|S)$ can be obtained analytically.

1) Tractable Marginal Likelihood - Direct optimization—When the full likelihood $p(\mathbf{y}|\theta, \mathbf{h}, S)$ and the parameter prior $p(\theta, \mathbf{h}|S)$ are in the conjugate family, the integral in (44) can be solved analytically. One common case of this kind consists of the linear Gaussian likelihood function with the Gaussian Inverse-Gamma prior [47]. The specific regulatory models that fall into this category are linear Gaussian models and Multinomial models in the gene expression layer. However, to obtain the MAP topology based on the GBScore, the optimization in (42) is NP hard and a numerical search must be applied for large networks.

A family of algorithms based on greedy hill climbing, including the K2 algorithm, is popular for searching the network topology. The general algorithm of greedy search is summarized in Figure 3. The greedy search algorithm performs local searches and includes a node into the network if the inclusion maximizes the (G)BScore. The greedy search algorithms in general are very fast with low computational burden. However, they can only guarantee a non-optimal local maximum. The problem can be relieved to a certain degree by invoking multiple independent searches with either random starts or the starts generated from the prior distribution of the topology. Tabu search list can be also introduced to record the previous

search path. The solution in the list will be avoided in the subsequent search and the algorithm terminates when no increase in the score is observed for a predefined number of successions. The use of Tabu search can further enhance the ability of the algorithm to escape the local maxima.

Simulated annealing (SA)[47] has also been applied for the network topology search. SA is a stochastic optimization algorithm aiming at obtaining the MAP solution by drawing samples. The summary of the algorithm is depicted in Figure 4. SA starts with constructing a distribution through Boltzmann's machine as

$$p_T(S) = Z^{-1} e^{GBScore(S)/T} \quad (45)$$

where Z is the normalizing constant and T is called the annealing temperature. Then, random samples of S are drawn from $p(S)$ based on the Metropolis-Hastings algorithm [47] with a sequence of decreasing T or a cooling procedure. For a large temperature T , samples are unlikely to be stuck at a local high density or essentially local maxima, thus driving the search more smoothly towards the global maxima. As the samples near the global optima, T is taken to be a low temperature, which restricts the search move to the local area of global optima and effectively locks down the optimal solution. Theoretically, SA can provide the global optimal solution. However, in practice, the optimality of solution depends largely on the cooling procedure and how to choose the best cooling procedure is still problematic. A rule of thumb has been provided in the literature. Multiple independent runs with different starts have also been proposed. In terms of computation, SA is much more demanding than greedy search algorithms.

2) Intractable marginalized likelihood—In almost all cases other than linear Gaussian and Multinomial expression models, no tractable marginalized likelihood can be obtained and numerical integration is thus needed in addition to the numerical topology search. Two approaches are commonly adopted depending on the way that numerical integration and topology search algorithms are combined.

The first approach, known as the Candidate method, approximates the (log) marginal likelihood by

$$\log p(\mathbf{y}|S) \approx \log p(\mathbf{y}|\hat{\theta}_S, \hat{\mathbf{h}}_S, S) - B(S) \quad (46)$$

where $\hat{\theta}_S$ and $\hat{\mathbf{h}}_S$ are the MAP estimate of θ and $\mathbf{h}$ under the topology S , and $B(S)$ is the penalty function. $B(S)$ is introduced to penalize the larger topologies and thus avoid overfitting. Three popular penalty functions are often applied, which are obtained based on the MDL criterion, the Akaike's information criterion (AIC), and Bayesian information criterion (BIC). They have the following respective expressions

$$AIC: \quad B(S) = k_{cd} \quad (47)$$

$$BIC: \quad B(S) = \frac{k_{cd}}{2} \log N \quad (48)$$

$$MDL: \quad B(S) = \frac{k_{cd}}{2} \log \frac{N}{2\pi} + \int |J(\theta, \mathbf{h})|^{1/2} d\theta d\mathbf{h} \quad (49)$$

where N is the number of data samples, k_{cd} is the combined dimension of $\hat{\theta}_S$ and $\hat{\mathbf{h}}_S$, and J is the Fisher information matrix. Since the second term is independent of N , MDL and BIC are asymptotically the same. Nevertheless, since N is relatively small in GRNs research, a performance difference exists between the two criterions [40]. The GBScore can be approximated by

$$GBScore(S) \approx \log p(\mathbf{y}|\hat{\theta}_S, \hat{\mathbf{h}}_S, S) - B(S) + \log p(S|\mathbf{d}). \quad (50)$$

To obtain the MAP solution that maximizes the GBScore, the search algorithms discussed above can be used. The main drawback of the Candidate method is that $\hat{\theta}_S$ and $\hat{\mathbf{h}}_S$ must be estimated for each topology that is visited in the topology search algorithms, thus making the implementation computationally expensive. Therefore, the Candidate approach is more appropriate for small networks or the networks that assume special structure such as tree.

When dealing with large networks, the second approach, known as structural expectation maximization algorithm (SEM) [15], alleviates the computational drawbacks of the Candidate method. Instead of obtaining the MAP parameter estimates for each possible topology, which often requires a numerical search, SEM includes a topology search inside an EM algorithm, where the E step is concerned with calculating an expectation of the unknown parameters and the M step performs a topology search. The algorithm is summarized in Figure 5.

Compared with the Candidate approach, where all possible network topologies are evaluated, SEM only concerns a limited topology realizations visited by the topology search in the M step. In the M step, any local search procedure as in the greedy hill climbing discussed in section V-A1 can be used. However, unlike the Candidate algorithm where the MAP parameters estimates are obtained for each evaluated topology realization, the expectation of the unknown parameter, $Q(S; S^{(n-1)})$, needs to be calculated. For the concerned nonlinear models including Gaussian model with spline interpolation, regression tree with Gaussian distribution, Sigmoidal Squash function, etc., evaluating $Q(S; S^{(n-1)})$ exactly is impossible. Therefore, the main computational difficulty in SEM is to calculate the expectation in the E step. Several different strategies can be adopted. For instance, numerical integration algorithms discussed in section V-B can be applied. However, computational complexity is a concern when choosing a numerical integration approach. Alternatively, the same method as the Candidate approaches can be employed when the integral is intractable, i.e., the integration is approximated by evaluating the integrand at the candidate estimates, such as the MAP estimates, of the unknowns. In this case, the computational gain of SEM can be perceived easily since only the MAP estimates for a much smaller size of topologies need to be calculated. Based on the theory of generalized EM, the computational complexity can be further reduced by selecting parameter estimates instead of MAP as long as Q function at the new estimates improves from the previous iteration. Note that, due to nonlinear regulatory function, numerical optimization is needed to calculate the MAP estimates. In the reduced complexity algorithm, the parameter estimates can be calculated with minimized complexity by avoid the complete iterative search steps needed to numerically evaluated the MAP estimates.

B. Probabilistic soft approaches

The difference between a probabilistic soft approach and a point-solution based hard approach is that the probabilistic approach aims to obtain an estimate of the APPs of the topology instead of just a point solution. The need for the APPs is two fold. First, the APPs provide a measurement on the confidence of inference. Secondly, the APPs are indispensable in a Bayesian approach for integrating multiple data sources, an important topic that is beyond the scope of this article. However, the computational complexity of probabilistic soft approaches is much higher than point-solution based approaches, restricting their use for large networks. To date, probabilistic soft approaches have mainly been applied to linear Gaussian expression models and Multinomial models [30], [25].

Under a Bayesian framework, the APPs of topology is obtained by

$$p(S|\mathbf{y}, \mathbf{d}) = \frac{p(\mathbf{y}|S)p(S|\mathbf{d})}{p(\mathbf{y})} \quad (53)$$

where $p(\mathbf{y}|s)$ is defined in (44) and $p(\mathbf{y})$ is the normalizing constant calculated by

$$p(\mathbf{y}) = \sum_{s \in \mathcal{S}} p(\mathbf{y}|S)p(S|\mathbf{d}) \quad (54)$$

Two difficulties prevent us from calculating the APPs exactly: First, as we discussed above, the marginal distribution $p(\mathbf{y}|S)$ can be intractable. Secondly, calculating the normalizing constant $p(\mathbf{y})$ is computationally prohibitive for large networks. To overcome the difficulties, numerical approaches are in place. We discuss in the following the two most popular approaches.

1) Markov chain Monte Carlo (MCMC) sampling—MCMC sampling [47] is a scheme designed based on Markov chains, which can generate random samples of S from the APPs. With the samples, the APPs can be approximated by

$$p(S|\mathbf{y}, \mathbf{d}) \approx \frac{1}{N} \sum_{j=1}^N S^{(j)} \quad (55)$$

where we assume that there are totally N samples and $S^{(j)}$ represents the j th samples. MCMC sampling generates these samples by constructing and simulating a Markov chain in the topology space with its stationary distribution equal to the APPs. When there are also unknown parameters, the Markov chain must move between the topology and the parameter space. The corresponding algorithm is called Reversible Jump MCMC (RJMCMC) sampling [47]. The summary of the RJMCMC algorithm is provided in Figure 6.

In the algorithm, the acceptance probability λ is calculated by

$$\lambda = \min \left\{ \frac{p(S^{(j)}, \theta^{(j)}, \mathbf{h}^{(j)}|\mathbf{y})q(S^{(j-1)}|\tilde{S})q(\tilde{\mathbf{u}}|\tilde{\theta}, \tilde{\mathbf{h}})}{p(S^{(j-1)}, \theta^{(j-1)}, \mathbf{h}^{(j-1)}|\mathbf{y})q(\tilde{S}|S^{(j-1)})q(\mathbf{u}|\theta^{(j-1)}, \mathbf{h}^{(j-1)})} \left| \frac{\partial g(\mathbf{u}, \theta^{(j-1)}, \mathbf{h}^{(j-1)})}{\partial \mathbf{u} \partial \theta^{(j-1)} \partial \mathbf{h}^{(j-1)}} \right|, 1 \right\}$$

The proposal distribution $q(\cdot|S^{(j-1)})$ determines how Markov chains move in the topology space. The popular birth-death RJMCMC scheme is a result of the uniform proposal distribution. By far, due the concern of complexity, only a special case of the RJMCMC algorithm has been applied in GRNs research. In this case, the conjugate priors are available for model parameters and unidentified variables; the marginalized likelihood $p(\mathbf{y}|S)$ can be then obtained analytically. Consequently, there is no sampling needed for parameters and unidentified variables. The corresponding RJMCMC is then a MH sampling in the topology space and it is also called the MCMC model composition (MC³) sampling. In [35], the conjugate Gaussian Inverse Gamma prior was used for a linear Gaussian model and in [30], [25] the conjugate Dirichlet prior was employed for a Binomial likelihood. In addition to the popular birth and death move, a swapping move, as used in [25], can be also included to improve the mixing of Markov chains.

2) Variational Bayesian Expectation Maximization—In general, Monte Carlo sampling algorithms are computationally too demanding, thus restricting their use only to small gene networks. For uncovering large networks, more efficient soft algorithms are needed. In what follows, we present a more efficient alternative called variational Bayesian Expectation Maximization (VBEM) algorithm.

The basic idea behind VBEM is to approximate the intractable posterior distributions over the parameters and topology with simpler tractable forms through optimizing a lower bound on the marginal likelihood $p(\mathbf{y})$ according to Jensen's inequality

$$\ln p(\mathbf{y}) \geq \ln \int \left[\sum_{S \in \mathcal{S}} q(S) \ln \frac{p(S, \mathbf{y} | \boldsymbol{\theta}, \mathbf{h})}{q(S)} + \ln \frac{p(\boldsymbol{\theta}, \mathbf{h} | \mathbf{y})}{q(\boldsymbol{\theta}, \mathbf{h})} \right] d\boldsymbol{\theta} d\mathbf{h} \quad (56)$$

where $q(\boldsymbol{\theta}, \mathbf{h})$ and $q(S)$ are the approximate distributions to be determined. The VBEM algorithm obtains these approximations by iteratively maximizing (56) with respect to $q(\boldsymbol{\theta}, \mathbf{h})$ and $q(S)$ according to the variational Bayesian learning rule [15] and mean field approximation. The general algorithm is summarized in Figure 7.

In [16], only the distributions of the model parameters $\boldsymbol{\theta}$ and the unidentified variables $\mathbf{h}$ are estimated in the VBEM algorithm, and it is demonstrated that by choosing conjugate exponential priors, both the VBE and the VBM steps can be calculated analytically. However, the VBM step becomes NP hard when $q(S)$ requires estimation. A solution is proposed in [35], which forces $q(s)$ to be a Gaussian distribution with the mean and the covariance matrix matched to the true APPs of S . As a result, in the VBE steps, the sum becomes the integration, which can be exactly carried out.

VI. Discussion on Biological Problems and Results from Applying the Surveyed Models

In this section, we discuss some of the biological problems that have been treated by the surveyed models. Before surveying the specific work, we want to point out that graphical models do not describe the actual interactions directly and thus there is no one-to-one correspondence between the topology of the real regulatory network and a graphical model used to describe it [26]. As a result, different graphical models can be applied to study the same regulatory network, for example, yeast cell cycle networks. This fact has two implications. First, there often needs a prior step to building graphical models that evaluate the association between actual regulatory networks and a potential graphical model and select a model based on the focus of a study. Different graphical models have different emphases and therefore are

limited in its only way. For instance, Gaussian Graphical Models do not have direction in its topology and cannot describe directional influence in real regulatory network. Also, Bayesian networks do not allow directed cycles, which cannot be used to model cyclic interactions in regulatory networks directly. However, the drawback of BNs might be overcome by using dynamic Bayesian networks. Therefore, the second implication is that there will need an additional step after network inference to translate the topology of the graphical model into that of the regulatory network. An example is provided in Figure 8, where in the left panel a DBN is used for the expression layer and red links are the inferred topology of the DBN, and in the right panel, the corresponding regulatory network is shown. It is clear that the two topologies are different. Also, even though there is no cycles in the DBN, a cycle between gene 2 and 3 exist in the gene network.

One of the targeted biology problems is to uncover the regulatory circuitry that controls cell cycles [5]. The cell cycle is the cycle of cell growth, DNA replication, and cell division and is tightly controlled through gene regulation and consists of several phases. Understanding the gene regulatory mechanism that controls cell cycles can shed light to cellular aging (senescence) and programmed cell death (apoptosis), processes known to be involved in the development of cancer and other diseases associated with the aging process. Work including [18], [42], [23], [43] have been devoted to studying yeast cell cycle networks; however, in [18], [42], [23], only microarray and gene expression layers were considered. Both discrete Bayesian networks and linear Gaussian model were applied to the gene expression layer and the fully correlated model was used for the microarray layer. The model was employed in a yeast cell cycle dataset containing 76 expression measures of 800 genes. In most of the tests, only 250 genes were considered. It is observed through the robustness analysis that the inferred network is sensitive to the specific adopted gene expression model. The discrete BNs and linear Gaussian model can result in different network topology; nevertheless, the resultant network still revealed interesting biological insights, for instance, a small number of dominant genes were identified. It is observed that many of the dominant genes are involved in controlling cell cycles. Also, the Markov analysis revealed the property of subnetworks: genes with high correlation tend to group together to form subnetworks and some of the genes serve to link the less correlated genes between subnetworks.

In [35], the cell cycle gene networks of malaria parasite *P. falciparum* were reconstructed using the gene sequence and microarray data. Unlike the aforementioned work on yeast where networks of hundreds of genes were covered based on limited data samples, a comparative genomic approach on sequence data is first applied to select only 38 proteins that are likely to participate in the cell cycle regulation. This selection makes the subsequent study of gene networks more reliable. To uncover the cell cycle gene network for the time series data, the DBN as shown in Figure 8 was used and the linear Gaussian model was adopted. To learn the network topology and model parameters jointly, a VBEM algorithm was developed. The uncovered network exhibits the common character of scale free network, and further investigation revealed several highly connected hub proteins that may be involved in kinase signaling cascade and transport/sorting processes. Moreover, a previously unidentified cyclin was predicted to regulate six downstream interactions. This finding is significant as cyclins are central regulators in cell cycle progression, but to date very few cyclins have been found in malaria parasite

In addition to using the microarray data in isolation, location data from [43] were considered in [42] as explanatory data for inferring yeast cell cycle gene networks. The data was produced from the location analysis of 114 yeast transcriptional factors. A p -value was generated for each TF to indicate the confidence of TF involvement in the regulation during cell cycles. As discussed in section IV-E, the p -values were translated to the prior distribution of topology. The network inference study considered only 25 genes, of which 10 were transcriptional

factors. To validate the results, a gold standard network was constructed based on existing biological facts. The results indicate that the integration of microarray data and location data can improve confidence of using either one data type alone. In [23], explanatory datasets were also considered. The prior distribution of network topology was first constructed according to [41] using the Gibbs random field based on multiple types of yeast datasets as discussed in IV-E. This prior was integrated together with the microarray data by using either linear Gaussian or dBN models for the gene expression layer; two networks were studied. The first one was derived from the KEGG pathway database and the second was a metabolic pathway. The inferred networks were shown to have a large overlap with the known biological networks, and this was considered a big improvement over the results obtained by the microarray data alone.

In [16], the networks of the human T-cell activation were studied in three layers: the microarray data, the gene expression, and an unidentified factor were considered for modeling; in particular, a fully correlated model was used for the microarray data. The specific graphical modeling framework is shown in Figure 9. To model the dynamics of each layer and the interaction between unidentified factors and gene expression, linear Gaussian model was applied, resulting in a linear Gaussian state space representation. The VBEM algorithm was developed to learn the parameters of the model, and the network topology was determined by thresholding the inferred weighting parameters. The inference was conducted on a set of highly replicated time series microarray profiles, consisting of 44 replicates of 10 time samples. It was observed that a large number of uncovered interactions were also represented by a previous study using the bootstrap method. In particular, a subnetwork including interactions between Jun-D and Jun-B appeared to be consistent with another proposed hypothesis of programmed cell death.

The reconstruction of the well known S.O.S. DNA Repair network of the *E. coli* bacterium based on microarray data was investigated in [19]. The specific graphical modeling framework is shown in Figure 10. This network is responsible for repairing DNA damage. For reconstruction, only 8 genes were selected for building the network, and the true network is also known. Microarray data from 4 experiments of different conditions were collected, with each having 50 evenly sampled expression levels over 6 minutes. A dynamic Bayesian network model was adopted, which consists of the microarray data layer and gene expression layer, where linear Gaussian models were chosen for both layers. An EM algorithm was developed to learn the parameters, and it showed that by imposing a regularization of sparse networks, the inferred network could fit the true network well.

In [17], perturbed expression data were used to infer the networks of yeast; in modeling, no unidentified factors were considered. A fully corrected model was used for the microarray data layer and a discrete Bayesian network was applied for the gene expression layer. The specific graphical modeling framework is shown in Figure 11. To learn the network topology, the GBScore function was calculated and a local score function was introduced to account for perturbation. A greedy search was applied for the purpose of network inference. The data for network construction consists of expression profiles of 565 genes obtained from 276 deletion mutants, 11 tetracycline regulatable alleles of essential genes, and 13 chemically treated *S. cerevisiae* cultures. Biological analysis showed many uncovered links supported by previous findings including, for example, steps in the *de novo* purine biosynthesis pathway and the mating signaling pathway. Another notable feature is that the algorithm can discover not only subnetworks that contains correlated genes but also inter-subnetwork interactions between weakly correlated genes.

PBN was applied to model gene regulatory networks in [2], [48], [28]. In all the work, the fully correlated model was chosen for microarray data layer and PBN was employed to model gene

regulations. In [48], based on the model, two subnetworks of glioma were grown out of two seed genes IGFBP2 and VEGF. The two seed genes were picked as they have been observed to overexpress in glioblastoma, the most advanced stage of brain tumors. Analysis of the two constructed subnetworks using prior knowledge and experiments offered many insights and supporting evidence. For instance, in the IGFBP2 based subnetwork, $\text{NF-}\kappa\text{B}$ is shown to activate IGFBP2, which can be confirmed by promoter sequence analysis and the existing literature. In the VEGF based subnetwork, an analysis of 5 genes in the network reflects their known function and physical relationship in a cell. In [28], a probabilistic neural network was used for the predictor of the PBN. A RJMCMC algorithm was applied to build the networks for malignant melanoma. Ten genes were selected from a set of 587 genes from the melanoma dataset and 31 microarray data samples were available for reconstruction. The steady state analysis showed that the steady-state distribution of the inferred network contains attractors that are either identical or very similar to the original observations. Moreover, many of the attractors in the network are singletons which resemble the biological definition of attractors.

In [31], [32], [34], GGMs were applied to different yeast microarray data sets in order to infer the genetic network. In modeling, no unidentified factors were considered and a fully correlated model was adopted for the microarray data layer in all these works. The specific graphical modeling framework is shown in Figure 11. A preprocessing step was done by performing clustering. In [31], the authors evaluated their approach in a galactose-utilization dataset to detect galactose-regulated genes in *S. cerevisiae*. In [34], the authors applied and tested their GGM models using microarray gene expression data by constructing a regulatory network of the 40 genes in the isoprenoid pathways in the plant *Arabidopsis thaliana*. 16 of those genes were assigned to the cytosolic MVA pathway, 19 assigned to the plastidal MEP pathway, and five genes encoding proteins located in the mitochondria. The results in [34] indicated that empirically defined associations based on the sparse Gaussian graphs, indeed, link to functional activities in isoprenoid metabolic pathways and many key biological interactions in the isoprenoid metabolic pathways were captured by the constructed networks. In [31], the authors constructed a genetic network and identified candidate genes for cross-talk between both MVA and MEP pathways. Interestingly, both positive and negative correlations were found between the identified candidate genes and the corresponding pathways. In [32], the authors developed an analysis of expression data from 158 breast cancer samples arising in their studies of molecular phenotyping for clinical prediction, and in particular, they were able to identify the genes of two pathways: the Estrogen receptor pathway and the Rb-E2F. The latter pathway is the key regulatory process governing the transition of cells from a quiescent state to a growing state while the former pathway plays key roles in breast cancer and the evolution and behavior of tumors.

VII. Conclusion

Statistical models were surveyed for reverse engineering GRNs in this paper. To draw connections between different existing models and present them from a system-level perspective, this survey was conducted on a graphical modeling framework. The framework has multiple connected layers, each of which defines a certain aspect of gene regulatory systems. Models were reviewed for each layer separately and discussed in the forms of conditional distributions. Conditional distributions enable a unified view among different models, making their connections and differences more obvious, and in addition, conditional distributions also serve as links between layers, through which a system-level understanding of models can be achieved. In Figure 12, we summarize the features of the surveyed models by layers. Except the noninformative priors, which are mainly designed to restrict the topology space, we delineated the pros and cons for each surveyed model. These models are not selected in isolation; instead, their position and influence in overall gene regulatory systems and their induced computational complexity in reverse engineering processes are considered. Depending

on the objective of an application, we might favor one model over the other. Nevertheless, from an accuracy standpoint, none of the surveyed models are sufficiently accurate to describe the true physics and biology of gene regulatory systems; most of the existing work relies on oversimplified models. Also, owing to the relative easy access of microarray expression data, efforts have been focused on expression layers. Many other aspects of gene regulatory systems have not yet been properly addressed, for instance, many different unidentified factors exist with differing functions in the system, but only coarse additive models have been used in the unidentified factors layer. Also, the nonstationary behavior of gene regulatory systems have been almost ignored in the current models.

The aforementioned problems in current models are caused by small data sample size. Including microarray data, only a very limited number of replicates are commonly available from a single experiment, making the use of complex and accurate models difficult, since reliable inference can hardly be guaranteed with limited data. The natural remedy to this problem is data integration. A trademark of post-genomic systems biology research is the ever-increasing accessibility to various biological data. A diverse collection of high-throughput biological data sources is currently available for elucidating GRNs; it would be advantageous to integrate them for network inference. Only by integrating all available data we can employ complex models and exploit more aspects of gene regulatory systems in modeling. Indeed, data integration has quickly emerged as a priority in the GRNs research [49]. Graphical modeling including Bayesian networks coupled with Bayesian inference theory is one of the most attractive paradigms for data integration, and in our GMF, we show how this can be achieved. We have reviewed a few models for integrating different types of data with microarray data. However, we have not yet reviewed the models for integrating multiple types of data. Increasingly more work has been published addressing the problem, but nevertheless it is still an emerging area, requiring much more effort from the computational communities, including signal processing.

Obstacles exist in computational techniques and computing power that prevent the use of complex models. We review many existing inference algorithms categorized by point and probabilistic solutions and we summarize in Figure 13 the surveyed algorithms. The majority of the existing work relies on point solutions, mainly due to their relatively low complexity; however, probabilistic solutions start to draw increasing attention as data integration is becoming a focus in this research. Even though MCMC sampling based algorithms can handle a great number of different models, computational complexity is still too high; also, their complexity is not scalable and thus their use on large networks is computationally infeasible. From a complexity perspective, the deterministic methods such as the VBEM are more appealing. However, the current VBEM solutions are limited to conjugated exponential families. Alternative algorithms including expectation propagation can be helpful in other cases; however, the current deterministic algorithms are in general restricted to limited scenarios. These facts call for continuing efforts from computational communities to develop more powerful and efficient inference algorithms.

As a final note, we feel encouraged from the recent involvement of the statistical signal processing community in GRNs research [2], [50]. Modeling and inference have always been the central themes of statistical signal processing research, which has a long and successful history of modeling complex systems. Because a repertoire of mathematical tools capable of making reliable inferential decisions and predictions is already available to statistical signal processing, we anticipate that a major impact will be noted in GRNs research. Reverse engineering GRNs also pose a host of unique problems and challenges; this research can ultimately enrich the theories and applications of statistical signal processing. This survey can help the signal processing community better understand development and future of reverse engineering GRNs, thus bringing statistical signal processing closer to the core of this research.

Yufei Huang received his Ph.D. degree in Electrical Engineering from the State University of New York at Stony Brook in 2001. Since 2002, he has been with the Department of Electrical and Computer Engineering at the University of Texas at San Antonio (UTSA), where he is now an Associate Professor. His current interests include data integration for gene networks discovery, context-based classification, miRNA targets and mass spectrometry data analysis. He was a recipient of National Science Foundation (NSF) CAREER award in 2005 and 2007 Best Paper Award of IEEE Signal Processing Magazine. His research has been supported by NSF, National Institute of Health, and Air Force Office of Scientific Research. He has served as an organizer of workshops and a guest editor of special issues in the area of genomic signal processing. He is an associate editor of EUROSP Journal on Bioinformatics and Computational Biology.

Yufeng Wang received her B.S. in Genetics from Fudan University, Shanghai, China, her M.S. in Statistics and Genetics, and her Ph.D. in Bioinformatics and Computational Biology from Iowa State University, Ames, IA. From 2001 to 2003, she was a research scientist at American Type Culture Collection (ATCC) and a research assistant professor at George Mason University, Manassas, VA. Since 2003, she has been with University of Texas at San Antonio, where she is an assistant professor with the Department of Biology. She is also an assistant professor at the South Texas Center for Emerging Infectious Diseases at San Antonio, Texas. Her current research interests include comparative genomics, molecular evolution, and systems biology, with a special emphasis on the evolutionary mechanisms of infectious diseases. Her research has been supported by National Institutes of Health, FDA, San Antonio Area Foundation, and University of Texas at San Antonio.

Isabel M. Tienda-Luna was born in Doña Mencía, Córdoba, Spain in 1978. She received her B.S. and M.S. degrees from the University of Granada in 1999 and 2001 respectively and her Ph.D. degree with honors in 2006. Her predoctoral research was founded by the “Ministerio de Educación y Ciencia” of Spain in the Systems, Signals and Waves group of the Applied Physics Department of the University of Granada. In 2007, she joined the Department of Electronics and Computer Science of the University of Granada as an Assistant Professor. Her research interests are in the areas of statistical signal processing and its application to system biology problems as well as simulation of semiconductor devices.

Acknowledgments

We thank the reviewers for their constructive comments.

This work is supported by an NSF Grant CCF-0546345 to Y. Huang. I. M. Tienda-Luna is supported by the Consejería de Innovación, Ciencia y Empresa (Junta de Andalucía) under projects P07-TIC-03269 and P07-TIC-02589. Y. Wang is partially supported by NIH grant 1R21AI067543-01 and NIH RCMI grant 2G12RR013646. The project described is also supported by grant number 1SC1GM081068 from the National Institute of General Medical Sciences to Y. Wang. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institute of General Medical Sciences, National Institute of Allergy and Infectious Diseases or the National Institutes of Health.

References

1. Friedman N. Inferring cellular networks using probabilistic graphical models. *Science* Feb;2004 303:799–805. [PubMed: 14764868]
2. Dougherty, ER.; Shmulevich, I.; Chen, J.; Wang, ZJ. *Genomic Signal Processing and Statistics*. Vol. 2. Hindawi Publishing Corporation; 2005. EURASIP Book Series on SP&C
3. Kyoda K, Baba K, Onami S, Kitano H. DBRF-MEGN method: an algorithm for deducing minimum equivalent gene networks from large-scale gene expression profiles of gene deletion mutants. *Bioinformatics* 2004;20:2662–2675. [PubMed: 15166016]

4. de Jong H. Modeling and simulation of genetic regulatory systems: A literature review. *Journal of Computational Biology* 2002;9(1):67–103. [PubMed: 11911796]
5. Alberts, B.; Bray, D.; Lewis, J.; Raff, M.; Roberts, K.; Watson, P. *Molecular Biology of the Cell*. Vol. 4. Garland Publishing; 2002.
6. Schena M, Shalon D, Davis PBRW. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science* 1995;270:467–70. [PubMed: 7569999]
7. Shaffer C. Next-generation sequencing outpaces expectations. *Nature Biotechnology* 2007;25:149.
8. Mikkelsen T, Ku M, Jaffe D, Issac B, Lieberman E, Giannoukos G, Alvarez P, Brockman W, Kim T, Koche R, et al. Genome-wide maps of chromatin state in pluripotent and lineage-committed cells. *Nature* 2007;448:553–560. [PubMed: 17603471]
9. Deplancke B, et al. A gene-centered *C. elegans* protein-DNA interaction network. *Cell* 2006;125(6):1193–1205. [PubMed: 16777607]
10. Gavin AC, et al. Proteome survey reveals modularity of the yeast cell machinery. *Nature* 2006;440(7084):631–636. [PubMed: 16429126]
11. Li S, et al. A map of the interactome network of the metazoan *C. elegans*. *Science* 2004;303(5657):540–543. [PubMed: 14704431]
12. Tong A, Evangelista M, Parsons A, Xu H, Bader G, Page N, Robinson M, Raghibizadeh S, Hogue C, Bussey H, et al. Systematic Genetic Analysis with Ordered Arrays of Yeast Deletion Mutants. *Science* 2001;294(5550):2364–2368. [PubMed: 11743205]
13. Gaur RK. RNA interference: a potential therapeutic tool for silencing splice isoforms linked to human diseases. *Biotechniques* 2006;Suppl:15–22. [PubMed: 16629383]
14. Kanehisa M, et al. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res* 2006;34:D354–357. [PubMed: 16381885]
15. Murphy, K. A brief introduction to graphical models and Bayesian networks. 1998. <http://www.cs.ubc.ca/m~urphyk/Bayes/bnintro.html>
16. Beal MJ, Falciani F, Ghahramani Z, Rangel C, Wild DL. A Bayesian approach to reconstructing genetic regulatory networks with hidden factors. *Bioinformatics* Sept;2004 20:1361–1372. [PubMed: 14962938]
17. Peér D, Regev A, Elidan G, Friedman N. Inferring subnetworks from perturbed expression profiles. *Bioinformatics* 2001;1(1):1–9.
18. Friedman N, Linial M, Nachman I, Pe’er D. Using Bayesian networks to analyze expression data. *Journal of Computational Biology* 2000;7(3–4):601–620. [PubMed: 11108481]
19. Perrin B, Ralaivola L, Mazurie AE, Bottani S, Mallet J, d’Alché Buc F. Gene networks inference using dynamic Bayesian networks. *Bioinformatics* 2003;19 Suppl 2:ii138–ii148. [PubMed: 14534183]
20. Sebastiani P, Gussoni E, Kohane IS, Ramoni M. Statistical challenges in functional genomics (with discussion). *Statistical Science* 2003;18(1):33–60.
21. Gat-Viks, I.; Tanay, A.; Raijman, D.; Shamir, R. The factor graph network model for biological systems. *Proceedings of RECOMB* 2005; 2005.
22. Segal, E. PhD dissertation. Stanford University; 2004. Rich probabilistic models for genomic data.
23. Kim SY, Imoto S, Miyano S. Inferring gene networks from time series microarray data using dynamic Bayesian networks. *Briefings in Bioinformatics* 2003;4(3):228–235. [PubMed: 14582517]
24. van Berlo RJP, van Someren EP, Reinders MJT. Studying the conditions for learning dynamic Bayesian networks to discover genetic regulatory networks. *Simulation* 2003;79:12.
25. Husmeier D. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics* 2003;19(17):2271–2282. [PubMed: 14630656]
26. Yeang TICH, Jaakkola T. Physical network models. *Journal of Computational Biology* 2004;11(2–3):243–263. [PubMed: 15285891]
27. Gat-Viks DR, Tanay A, Shamir R. A probabilistic methodology for integrating knowledge and experiments on biological networks. *Journal of Computational Biology* 2006;13(2):165–181. [PubMed: 16597233]

28. Zhou X, Wang X, Pal R, Ivanov I. A Bayesian connectivity-based approach to constructing probabilistic gene regulatory networks. *Bioinformatics* 2004;20(17):2918–2927. [PubMed: 15145802]
29. Lahdesmaki H, Hautaniemi S, Shmulevich I, Yli-Harja O. Relationships between probabilistic Boolean networks and dynamic Bayesian networks as models of gene regulatory networks. *Signal Processing* April;2006 86(4):814–834. [PubMed: 17415411]
30. Bulashevskaya S, Eils R. Inferring genetic regulatory logic from expression data. *Bioinformatics* 2005;21(11):2706–2713. [PubMed: 15784747]
31. Wille A, Zimmermann P, Vranova E, Furholz A, Laule O, Bleuler S, Henning L, Prelic A, Rohr PV, Thiele L. Sparse graphical Gaussian modeling of the isoprenoid gene network in *Arabidopsis thaliana*. *Genome Biology* 2004;5:1–13.
32. Dobra A, Jonnes B, Hans B, Nevis C, West M. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis* 2004;90:196–212.
33. de la Fuente A, Bing N, Hoeschele I, Mendes P. Discovery of meaningful associations in genomic data using partial correlation coefficients. *Bioinformatics* 2004;20(18):3565–3574. [PubMed: 15284096]
34. Li H, Gui J. Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks. *BioStatistics* 2006;7(2):302–317.
35. Luna IT, Yin Y, Huang Y, Perez MCC, Cai H, Sanchez M, Wang Y. Inferring the skeleton cell cycle regulatory network of malaria parasite using comparative genomic and variational Bayesian approaches. *Genetica* 2008;132(2)
36. Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, Favera RD, Califano A. Aracne: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* 2006;7(Suppl S7)
37. de Hoon, M.; Imoto, S.; Miyano, S. *Lecture Notes in Computer Science*. Vol. 2534. Springer; Berlin/Heidelberg: 2002. p. 267-274.ch. Inferring Gene Regulatory Networks from Time-Ordered Gene Expression Data Using Differential Equations
38. Wang ZJ, Chen J, Liu KJR. Quantitative modeling of genetic regulatory networks by incorporating genomic data sources. *IEEE workshop on Genomic Signal Processing and Statistics*. June;2004
39. Friedman, N.; Goldszmidt, M. *Graphical Modeling*. Vol. 1. Vol. VI. MIT Press; Kluwer Academic: 1998. Learning BNs with local structure, in *Learning in Graphical Models*; p. 421-459.
40. Zhou X, Wang X, Dougherty E. Gene selection using logistic regressions based on AIC, BIC and MDL criteria. *New Mathematics and Natural Computation* 2005;1(1):129–145.
41. Imoto, S.; Higuchi, T.; Goto, T.; Tashiro, K.; Kuhara, S.; Miyano, S. Combining microarrays and biological knowledge for estimating gene networks via Bayesian networks; *Proceedings of the Computational Systems Bioinformatics*; 2003.
42. Allister Bernard AJH. Informative structure priors: Joint learning of dynamic regulatory networks from multiple types of data. *Proceedings of the Pacific Symposium on Biocomputing*. 2005
43. Lee TI, Rinaldi NJ, Robert F, et al. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science Magazine* Oct;2002 298:799–804.
44. di Bernardo D, Gardner TS, Collins JJ. Robust identification of large genetic networks. *Proceedings of the Pacific Symposium on Biocomputing* 2004;9:486–497.
45. Segal E, Wang H, Koller D. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics* 19(S1) (Proc ISMB), 2003, winner of the ISMB Best Student Paper Award.
46. Segal, E.; Koller, D. Probabilistic hierarchical clustering for biological data. *Sixth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*; Washington, DC. April 2002; p. 273-280.
47. Robert, CP.; Casella, G. *Monte Carlo Statistical Methods*. Vol. 2. Springer; 2004.
48. Hashimoto F, Kim S, Shmulevich I, Zhang W, Bittner ML, Dougherty ER. Growing genetic regulatory networks from seed genes. *Bioinformatics* 2004;20(8):1241–1247. [PubMed: 14871865]
49. Myers, CL.; Robson, D.; Wible, A.; Hibbs, MA.; Chiriac, C.; Theesfeld, CL.; Dolinski, K.; Troyanskaya, OG. *Genome Biology*. Vol. 6. 2005. Discovery of biological networks from diverse functional genomic data; p. R114

50. Dougherty, ER.; Astola, J.; Chen, J.; Goutsias, J.; Shmulevich, I. Guest editorial special issue on Genomic Signal Processing; IEEE Transactions on Signal Processing; June; 2006.

Fig. 1.
An illustration of gene regulation.

Fig. 2.
The proposed graphical modeling framework for gene regulation

The Greedy Search Algorithm

Choose an initial topology $S^{(0)}$;

For $n = 1, 2$, until converge.

Calculate $S^{(n)} = \arg \max_{S \in \mathcal{S}_{n-1}} GBScore(S)$ where

$\mathcal{S}_{n-1}$ contains the neighboring topologies of $S^{(n-1)}$;

Return $S^{(n)}$ and terminate if $S^{(n)} = S^{(n-1)}$

;

Fig. 3.

The Greedy Search Algorithm

Fig. 4.
The Simulated Annealing Algorithm

Fig. 5.
The Structural EM Algorithm

Algorithm: RJMCMC

Provide a set of initial samples $S^{(0)}$, $\theta^{(0)}$, and $\mathbf{h}^{(0)}$.

Iterate for $j = 1 : J$

Choose a new topology $\tilde{S}$ from a proposal distribution $q(\cdot | S^{(j-1)})$

Draw $\mathbf{u}$ from the trial distribution $q(\cdot | \theta^{(j-1)}, \mathbf{h}^{(j-1)})$.

Calculate the mapping $g(\cdot)$ as $(\tilde{\mathbf{u}}, \tilde{\theta}, \tilde{\mathbf{h}}) = g(\mathbf{u}, \theta^{(j-1)}, \mathbf{h}^{(j-1)})$.

With probability λ accept $S^{(j)} = \tilde{S}$, $\theta^{(j)} = \tilde{\theta}$ and $\mathbf{h}^{(j)} = \tilde{\mathbf{h}}$

Otherwise Set $S^{(j)} = S^{(j-1)}$, $\theta^{(j)} = \theta^{(j-1)}$ and $\mathbf{h}^{(j)} = \mathbf{h}^{(j-1)}$.

Fig. 6.
The Reversible Jump MCMC Algorithm

The VBEM Algorithm

Start with initial distributions $q^{(0)}(\boldsymbol{\theta}, \mathbf{h})$ and $q^{(0)}(S)$

Iterate for $n = 0, 1, 2 \dots$ until converge

VBE step:

$$q^{(n+1)}(S) = \frac{1}{Z_S} \exp\left[\int q^{(n)}(\boldsymbol{\theta}, \mathbf{h}) \ln p(S, \mathbf{y}|\boldsymbol{\theta}, \mathbf{h}) d\boldsymbol{\theta} d\mathbf{h}\right] \quad (57)$$

VBM step:

$$q^{(n+1)}(\boldsymbol{\theta}, \mathbf{h}) = \frac{1}{Z_{\boldsymbol{\theta}, \mathbf{h}}} p(\boldsymbol{\theta}, \mathbf{h}) \exp\left[\sum_{S \in \mathcal{S}} S q^{(n+1)}(S) \ln p(S, \mathbf{y}|\boldsymbol{\theta}, \mathbf{h})\right]. \quad (58)$$

Fig. 7.
The VBEM Algorithm

Fig. 8.

The left side panel shows a DBN model as used in [35]. The right panel demonstrates the corresponding regulatory network described by the DBN.

Fig. 9.
The corresponding graphical modeling framework for gene regulation in [16].

Fig. 10.

The corresponding graphical modeling framework for gene regulation in [19]

Fig. 11.

The corresponding graphical modeling framework for gene regulation with in [17] and GGMs.

A. Microarray Data	B. Gene Expression		C. Hidden Factors	D. Parameters	E. Prior Evidence
<p>A1. <u>Fully correlated model</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simplest • Drawbacks: Oversimplified <p>A2. <u>Linear Gaussian model</u></p> <ul style="list-style-type: none"> • Advantages: Low computation; Captures the stochastic phenomena • Drawbacks: Unrealistic since the microarray is nonlinear and nonGaussian in nature <p>A3. <u>Gaussian model with discrete expression levels</u></p> <ul style="list-style-type: none"> • Advantages: Captures nonlinear and nonGaussian of data • Drawbacks: Relatively high computation; unrealistic discrete expression levels; unrealistic Gaussian assumption. 	<p>B1. <u>Dyn. Bayesian Net-Multinomial Model</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simple; captures nonlinearity in gene expression • Drawbacks: Lost of information from discretization <p>B2. <u>Factor Graph – discrete model</u></p> <ul style="list-style-type: none"> • Advantages: Allow both cyclic and undirected links • Drawbacks: Lost of information <p>B4. <u>Boolean-logic Function</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simple; models specific regulatory functions • Drawbacks: Lost of information from discretization <p>B6. <u>Linear Gaussian model</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simple; models continuous expression levels; • Drawbacks: Does not model nonlinear regulation <p>B8. <u>Gaussian model with Spline interpolation</u></p> <ul style="list-style-type: none"> • Advantages: Models nonlinear regulation; low computational complexity • Drawbacks: Depends on the basis functions; can be unrealistic <p>B10. <u>Regression tree with Gaussian distribution</u></p> <ul style="list-style-type: none"> • Advantages: models nonlinear regulation; • Drawbacks: Require additional evidence in building the tree; high computational complexity 	<p>B3. <u>Probabilistic Boolean Networks</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simple; models specific regulatory functions; captures nonlinearity in gene regulation • Drawbacks: Lost of information from discretization <p>B5. <u>Graph Gauss. Model</u></p> <ul style="list-style-type: none"> • Advantages: handle undirected loops; models continuous expression levels; • Drawbacks: difficult for high-dimensional data; can't capture combinatorial effects <p>B7. <u>Gaussian model with Sigmoidal Squash function</u></p> <ul style="list-style-type: none"> • Advantages: Models nonlinear saturation effect of regulation; • Drawbacks: High computational complexity <p>B9. <u>Gaussian Kernel</u></p> <ul style="list-style-type: none"> • Advantages: Nonlinear nonGaussian Modeling • Drawbacks: Relies on data and thus can be unrealistic, high complexity <p>B11. <u>Linear Gaussian Model with Non-uniform Sampling Interval</u></p> <ul style="list-style-type: none"> • Advantages: Models non-uniform sampled data • Drawbacks: Unrealistic regulation model 	<p>C Linear Gaussian</p> <ul style="list-style-type: none"> • Advantages: computational simple • Drawbacks: less accurate 	<p>D1. <u>Dynamics of model coefficient: linear Gaussian data</u></p> <ul style="list-style-type: none"> • Advantages: captures the nonstationarity of the network • Drawbacks: Unrealistic, high computational complexity <p>D2. <u>Dynamics of network topology: Stationary model</u></p> <ul style="list-style-type: none"> • Advantages: Simply and reasonable for experiments of short duration. • Drawbacks: Unrealistic for systems with functional or phenotype changes 	<p>E1. <u>Noninformative priors</u></p> <ul style="list-style-type: none"> • Power law: Captures the scale invariance property of the biological networks; • MDL: Encourages networks with small connectivity; • Independent Bernoulli prior: Assumes independent regulation. <p>E2.A <u>The Gibbs random field</u></p> <ul style="list-style-type: none"> • Advantages: Models the joint distribution of network directly; Useful in modeling deterministic evidence; • Drawbacks: Computationally difficulty; Does not explicitly map the quantitative uncertainties in prior data <p>E2. B <u>Independent Bernoulli – Mapping p values</u></p> <ul style="list-style-type: none"> • Advantages: Computationally simple; provide a way to map p values in prior data. • Drawbacks: Does not model dependence between edges directly. <p>E2. C <u>Markov random field</u></p> <ul style="list-style-type: none"> • Advantages: flexible and model protein interaction and gene modules • Drawbacks: can't model directed relationship

Fig. 12.
Summary of surveyed models.

Point Solutions - Generalized Bayesian Score				Probabilistic Solutions	
Tractable Marginalized Likelihood		Intractable Marginalized Likelihood			
Hill Climb Search	Simulated Annealing	Candidate methods	Structural EM	MCMC	VBEM
<ul style="list-style-type: none"> • <u>Advantages</u>: Low complexity • <u>Drawbacks</u>: Local optimal 	<ul style="list-style-type: none"> • <u>Advantages</u>: possible global optimal • <u>Drawbacks</u>: high complexity; 	<ul style="list-style-type: none"> • <u>Advantages</u>: possible global optimal for small networks • <u>Drawbacks</u>: high complexity 	<ul style="list-style-type: none"> • <u>Advantages</u>: lower complexity than candidate methods • <u>Drawbacks</u>: local optimal 	<ul style="list-style-type: none"> • <u>Advantages</u>: Applicable to general problems; good performance • <u>Drawbacks</u>: Very high complexity; does not scale well for large networks 	<ul style="list-style-type: none"> • <u>Advantages</u>: much lower complexity than MCMC; applicable to large networks • <u>Drawbacks</u>: restricted to limited models

Fig. 13.
Summary of surveyed algorithms.