



## OPEN ACCESS

## EDITED BY

Mahyar Fazlyab,  
Johns Hopkins University, United States

## REVIEWED BY

Salar Fattahi,  
University of Michigan, United States  
Valentina Breschi,  
Politecnico di Milano, Italy

## \*CORRESPONDENCE

Mohammad Alali,  
alali.m@northeastern.edu

## SPECIALTY SECTION

This article was submitted to AI and Machine Learning Control, a section of the journal Frontiers in Control Engineering

RECEIVED 11 August 2022

ACCEPTED 15 November 2022

PUBLISHED 13 December 2022

## CITATION

Alali M and Imani M (2022), Inference of regulatory networks through temporally sparse data.  
*Front. Control. Eng.* 3:1017256.  
doi: 10.3389/fcteg.2022.1017256

## COPYRIGHT

© 2022 Alali and Imani. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

# Inference of regulatory networks through temporally sparse data

Mohammad Alali\* and Mahdi Imani

Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, United States

A major goal in genomics is to properly capture the complex dynamical behaviors of gene regulatory networks (GRNs). This includes inferring the complex interactions between genes, which can be used for a wide range of genomics analyses, including diagnosis or prognosis of diseases and finding effective treatments for chronic diseases such as cancer. Boolean networks have emerged as a successful class of models for capturing the behavior of GRNs. In most practical settings, inference of GRNs should be achieved through limited and temporally sparse genomics data. A large number of genes in GRNs leads to a large possible topology candidate space, which often cannot be exhaustively searched due to the limitation in computational resources. This paper develops a scalable and efficient topology inference for GRNs using Bayesian optimization and kernel-based methods. Rather than an exhaustive search over possible topologies, the proposed method constructs a Gaussian Process (GP) with a topology-inspired kernel function to account for correlation in the likelihood function. Then, using the posterior distribution of the GP model, the Bayesian optimization efficiently searches for the topology with the highest likelihood value by optimally balancing between exploration and exploitation. The performance of the proposed method is demonstrated through comprehensive numerical experiments using a well-known mammalian cell-cycle network.

## KEYWORDS

topology inference, maximum likelihood estimation, gene regulatory networks, Boolean dynamical systems, Bayesian optimization

## 1 Introduction

Gene regulatory networks (GRNs) play an important role in the molecular mechanism of underlying biological processes, such as stress response, DNA repair, and other mechanisms involved in complex diseases such as cancer. The topology inference of GRNs is critical in systems biology since it can generate valuable hypotheses to promote further biological research. Furthermore, a deep understanding of these biological processes is key in diagnosing and treating many chronic diseases. Advances in high-throughput genomic and proteomic profiling technologies have provided novel platforms for studying genomics. Meanwhile, single-cell gene-expression measurements allow capturing multiple snapshots of these complex biological processes. These advances offer an opportunity for seeking systematic approaches to understand the structure of GRNs.

In recent years, Boolean network models have been successfully employed for modeling different biological networks (Wynn et al., 2012; Saadatpour and Albert, 2013; Abou-Jaoudé et al., 2016). More specifically, these Boolean networks have been widely used for inferring GRNs from their state (i.e., gene) data (Pusnik et al., 2022). The state value of genes in the Boolean network is represented by 1 and 0, representing the activation and inactivation of genes, respectively. There are several Boolean network models, including the deterministic Boolean network models, Boolean network with perturbation, probabilistic Boolean network models, and Boolean control networks (Lähdesmäki et al., 2003; Shmulevich and Dougherty, 2010; Cheng and Zhao, 2011). Most of these models account for genes' stochasticity and can effectively capture the dynamics of GRNs through relatively small times-series data.

Inference of Boolean network models consists of learning the parameters of their models given all the available data. Several advances have been made in the inference of Boolean network models in recent years. These techniques aim to find models that best fit the available time-series data. The fitness criteria are often likelihood or posterior, leading to well-known maximum likelihood and maximum a posteriori inference techniques (Shmulevich et al., 2002; Lähdesmäki et al., 2003). Despite the optimality of these inference techniques, lack of scalability has limited their applications to small GRNs. Several heuristic methods have been developed to scale the inference of Boolean network models; these include scale-free and cluster-based approaches (Hashimoto et al., 2004; Barman and Kwon, 2017), and methods built on evolutionary optimization techniques (Tan et al., 2020; Barman and Kwon, 2018). The former methods aim to build a topology from known seed nodes according to multiple heuristics, whereas the latter ones use evolutionary optimization techniques such as genetic algorithms and particle swarm algorithms for searching over the parameter space. Despite the scalability of these approaches, their incapability to effectively consider the temporal changes in data and efficiently search over possible network models leads to their unreliability in the inference process.

This paper focuses on developing a systematic approach for the inference of GRNs using Boolean network models. Two main challenges in the inference of GRNs are:

- *Large Topology Candidate Space:* The modeling consists of estimating a large number of interacting parameters, which represent the connections between genes that govern their dynamics. This requires searching over a large number of topology candidates and picking the one with the highest likelihood value given the available data. Most existing inference methods for general nonlinear models are developed to deal with continuous parameter spaces, such as maximum likelihood (Johansen et al., 2008; Kantas et al., 2015; Imani and Braga-Neto, 2017; Imani et al., 2020), expectation maximization (Hürzeler and Künsch, 1998; Godsill et al., 2004; Schön et al., 2011;

Wills et al., 2013), and multi-fidelity (Imani et al., 2019; Imani and Ghoreishi, 2021) methods. However, these methods cannot be applied for inference over large discrete parameter spaces, such as the large topology candidate space of GRNs. In this paper, we develop a method that is scalable with respect to the number of unknown interactions, and efficiently searches over the large topology candidate space. More specifically, our proposed method enables optimal inference in the presence of a large number of unknown regulations for GRNs with a relatively small number of genes.

- *Expensive Likelihood Evaluation:* The likelihood function, which measures the probability that the available data come from each topology candidate, is often expensive to evaluate. The reasons for that are a large number of genes in GRNs, and the sparsity in the data, which require propagation of the system stochasticity across time and gene states. Given the limitation in the computational resources, evaluation of the likelihood functions for all of the topology candidates is impossible, and one needs to find the topology with the highest likelihood value with a few expensive likelihood evaluations.

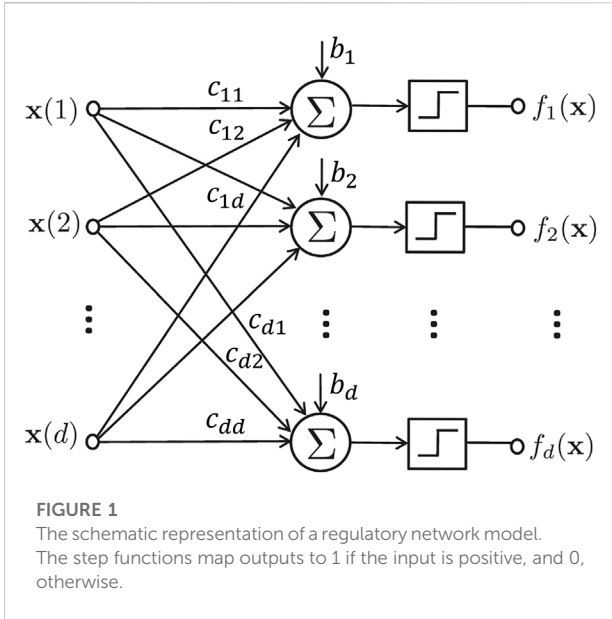
This paper derives a scalable topology inference for GRNs observed through temporally sparse data. The proposed framework models the expensive-to-evaluate (log-)likelihood function using a Gaussian Process (GP) regression with a structurally-inspired kernel function. The proposed kernel function exploits the structure of GRNs to efficiently learn the correlation over the topologies, and enables Bayesian prediction of the log-likelihood function for all the topology candidates. Then, a sample-efficient search over topology space is achieved through a Bayesian optimization policy, which sequentially selects topologies for likelihood evaluation according to the posterior distribution of the GP model. The proposed method optimally balances exploration and exploitation, and searches for the global solution without getting trapped in the local solutions. The accuracy and robustness of the proposed framework are demonstrated through comprehensive numerical experiments using a well-known mammalian cell-cycle network.

The remainder of this paper is organized as follows. Section 2 provides a detailed description of the GRN model and the topology inference of GRNs. Further, the proposed topology optimization framework is introduced in Section 3. Section 4 presents various numerical results, and the main conclusions are discussed in Section 5.

## 2 Preliminaries

### 2.1 GRN model

This paper employs a Boolean network with perturbation (BNp) model for capturing the dynamics of GRNs (Shmulevich



and Dougherty, 2010; Imani et al., 2018; Hajiramezani et al., 2019). Previously, several works have successfully employed the BNp model for different purposes such as inference (Dougherty and Qian, 2013; Marshall et al., 2007) and classification (Karbalaighareh et al., 2018). This model properly captures the stochasticity in GRNs, coming from intrinsic uncertainty or unmodeled parts of the systems. Consider a GRN consisting of  $d$  genes. The state process can be expressed as  $\{X_k; k = 0, 1, \dots\}$ , where  $X_k \in \{0,1\}^d$  represents the activation/inactivation state of the genes at time  $k$ . The gene state is updated at each discrete time through the following Boolean signal model:

$$X_k = f(X_{k-1}) \oplus n_k, \tag{1}$$

for  $k = 1, 2, \dots$ , where  $n_k \in \{0,1\}^d$  is Boolean transition noise at time  $k$ , “ $\oplus$ ” indicates component-wise modulo-2 addition, and  $f$  represents the network function.

The network function in Eq. 1 is expressed in component form as  $f = (f_1, \dots, f_d)$ . Each component  $f_i: \{0,1\}^d \rightarrow \{0, 1\}$  is a Boolean function given by:

$$f_i(\mathbf{x}) = \begin{cases} 1, & \sum_{j=1}^d c_{ij}\mathbf{x}(j) + b_i > 0, \\ 0, & \sum_{j=1}^d c_{ij}\mathbf{x}(j) + b_i \leq 0, \end{cases} \tag{2}$$

for  $i = 1, \dots, d$ , where  $c_{ij}$  denotes the type of regulation from component  $j$  to component  $i$ ; it takes +1 and -1 values if there is a positive and negative regulations from component  $j$  to component  $i$  respectively, and 0 if component  $j$  is not an input to component  $i$ .  $b_i$  is a tie-breaking parameter for component  $i$ ; it takes  $\frac{+1}{2}$  if an equal number of positive and negative inputs lead to state value +1 and reverse for  $\frac{-1}{2}$ . The network function in Eq. 2 can also be expressed in matrix form as:

$$f(X_{k-1}) = \overline{C}X_{k-1} + \mathbf{b}, \tag{3}$$

where the threshold operator  $\bar{v}$  maps the positive elements of vector  $v$  to 1 and negative elements to 0,  $C$  is the connectivity matrix with  $(C)_{ij} = c_{ij}$  in the  $i$ th row and  $j$ th column, and  $\mathbf{b} = [b_1, \dots, b_d]^T$  represents the bias vector. A schematic representation of the regulatory network model is shown in Figure 1.

In Eq. 1, the noise process  $n_k$  indicates the amount of stochasticity in a Boolean state process. For example,  $n_k(j) = 1$ , means that the  $j$ th gene’s state at time step  $k$  is flipped and does not follow the Boolean function. Whereas,  $n_k(j) = 0$  indicates that this state is governed by the network function. We assume that all the  $n_k$  components are independent and have a Bernoulli distribution with parameter ( $p$ ), which  $0 \leq p < 0.5$  refers to the amount of stochasticity in each state variable (i.e., gene).

## 2.2 Topology inference of regulatory networks

In practice, the network function is unknown or partially known, and the unknown parameters need to be inferred through available data. The unknown information is often the elements of the connectivity matrix or bias units. We assume that  $L$  elements of the connectivity matrix  $\{c^1, \dots, c^L\}$  are unknown. Given that each element takes in values in space  $\{+1, 0, -1\}$ , there will be  $3^L$  different possible models (i.e., connectivity matrices) denoted by parameter vectors:  $\Theta = \{\theta^1, \dots, \theta^{3^L}\}$ , where  $\theta^j = [\theta^j(1), \dots, \theta^j(L)]$ , and  $\theta^j(i)$  denotes the type of the  $i$ th unknown interaction/parameter under the  $j$ th model. It is evident that each parameter vector corresponds to one specific topology/model; therefore, the phrases parameter vector and topology/model are used interchangeably throughout this paper. Further,  $C^{\theta^j}$  represents the connectivity matrix associated with parameter vector  $\theta^j$ , while only one parameter vector represents the true underlying system topology. Assuming that  $D_{1:T}$  represents the available data, the inference process can be formulated as:

$$\theta^* = \underset{\theta \in \Theta}{\operatorname{argmax}} P(D_{1:T} | \theta), \tag{4}$$

where  $P(D_{1:T} | \theta)$  is the likelihood function for the topology parameterized by  $\theta$ . The solution to the optimization problem,  $\theta^*$  in Eq. 4, is known as the maximum likelihood solution. Note that without loss of generality, the proposed method, which will be described in the next section, can be applied to any arbitrary point-based estimator, such as maximum a posteriori.

It should be noted that the unknown parameters could include the bias units in the network model in Eq. 3. Depending on the regulatory network, the bias units are often  $\frac{-1}{2}$  for the network in normal conditions, whereas a combination of  $\frac{+1}{2}$  and  $\frac{-1}{2}$  often represents the network in stress conditions. Therefore, if the network condition is not known, the topology

inference could aim to simultaneously estimate the parameters of the connectivity matrix and bias units.

### 3 The proposed framework

#### 3.1 Likelihood evaluation

Let  $\{\mathbf{x}^1, \dots, \mathbf{x}^{2^d}\}$  be an arbitrary enumeration of the possible Boolean state vectors (i.e., a GRN with  $d$  components). The available data in  $D_{1:T}$  can be represented using the vector  $I_{1:T} = \{I_1, \dots, I_T\}$ , where  $I_k$  specifies the index associated with the  $k$ th state (0 if the state at time step  $k$  is missing). For instance,  $D_{1:6} = \{\tilde{\mathbf{X}}_2 = \mathbf{x}^9, \tilde{\mathbf{X}}_3 = \mathbf{x}^3, \tilde{\mathbf{X}}_6 = \mathbf{x}^{11}\}$  contains the information of time steps 2, 3 and 6, and denotes that data at time steps 1, 4 and 5 are missing. In this case, the indicator vector is defined as  $I_{1:6} = \{0, 9, 3, 0, 0, 11\}$ .

For any given model  $\theta \in \Theta$ , we define the predictive posterior distribution ( $\Pi_{k|k-1}^\theta$ ) and posterior distribution ( $\Pi_{k|k}^\theta$ ) of the states as:

$$\begin{aligned} \Pi_{k|k-1}^\theta(i) &= P(\mathbf{X}_k = \mathbf{x}^i | I_{1:k-1}, \theta), \\ \Pi_{k|k}^\theta(i) &= P(\mathbf{X}_k = \mathbf{x}^i | I_{1:k}, \theta), \end{aligned} \tag{5}$$

for  $i = 1, \dots, 2^d$  and  $k = 1, 2, \dots$

We define the *transition matrix*  $M^\theta$  of size  $2^d \times 2^d$  associated with a GRN model parameterized by  $\theta$ , through the following notation:

$$\begin{aligned} (M^\theta)_{ij} &= P(\mathbf{X}_r = \mathbf{x}^i | \mathbf{X}_{r-1} = \mathbf{x}^j, \theta) \\ &= P(\mathbf{n}_r = \mathbf{f}^\theta(\mathbf{x}^j) \oplus \mathbf{x}^i) \\ &= p^{\|\mathbf{C}^\theta \mathbf{x}^j + \mathbf{b}^\theta \oplus \mathbf{x}^i\|_1} (1-p)^{d - \|\mathbf{C}^\theta \mathbf{x}^j + \mathbf{b}^\theta \oplus \mathbf{x}^i\|_1}, \end{aligned} \tag{6}$$

for  $i, j = 1, \dots, 2^d$ , where the second and third lines in Eq. 6 are obtained based on the GRN model in Eq. 1.

Let  $\Pi_{0|0}^\theta(i) = P(\mathbf{X}_0 = \mathbf{x}^i | \theta)$ , for  $i = 1, \dots, 2^d$ , be the initial state distribution. If no knowledge about this distribution is available, this can be represented by  $\Pi_{0|0}^\theta(i) = 1/2^d$ , for  $i = 1, \dots, 2^d$ , and  $\theta \in \Theta$ . The predictive posterior distribution can be computed recursively as:

$$\Pi_{k|k-1}^\theta = M^\theta \Pi_{k-1|k-1}^\theta. \tag{7}$$

The posterior probability of states at time step  $k$  can be computed according to predictive posterior and the available data at time step  $k$ . If the data at time step  $k$  is missing, i.e.  $I_k = 0$ , the predictive posterior becomes the posterior, as no data is available at time step  $k$ . This can be written as:

$$\begin{aligned} \Pi_{k|k}^\theta(j) &= P(\mathbf{X}_k = \mathbf{x}^j | I_{1:k}, \theta) = P(\mathbf{X}_k = \mathbf{x}^j | I_{1:k-1}, I_k = 0, \theta) \\ &= \Pi_{k|k-1}^\theta(j), \text{ for } j = 1, \dots, 2^d. \end{aligned} \tag{8}$$

However, if the  $i$ th state is observed at time step  $k$ , i.e.,  $I_k = i$ , then the posterior probability of state at time step  $k$  becomes 1 for state

$i$ , as full knowledge about  $\mathbf{X}_k = \mathbf{x}^i$  is available. The posterior probability in this case can be expressed as:

$$\begin{aligned} \Pi_{k|k}^\theta(i) &= P(\mathbf{X}_k = \mathbf{x}^i | I_{1:k-1}, I_k = i, \theta) = 1, \\ \Pi_{k|k}^\theta(j) &= P(\mathbf{X}_k = \mathbf{x}^j | I_{1:k-1}, I_k = i, \theta) = 0, \text{ } j \neq i. \end{aligned} \tag{9}$$

To summarize, the posterior portability of any state at time  $k$ , i.e.  $\mathbf{X}_k = \mathbf{x}^i$  can be derived through the following expression:

$$\Pi_{k|k}^\theta(i) = \begin{cases} \Pi_{k|k-1}^\theta(i) & \text{if } I_k = 0, \\ 1 & \text{if } I_k = i, \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

for  $i = 1, \dots, 2^d$  and  $k = 1, 2, \dots$

The likelihood value in optimization problem in Eq. 4 can be written in logarithmic format as:

$$\begin{aligned} L(\theta) &:= \log P(D_{1:T} | \theta) = \log P(I_{1:T} | \theta) \\ &= \sum_{k=1}^T \log P(I_k | I_{1:k-1}, \theta), \end{aligned} \tag{11}$$

where

$$P(I_k | I_{1:k-1}, \theta) = \begin{cases} \Pi_{k|k-1}^\theta(I_k) & \text{if } I_k \neq 0 \\ 1 & \text{otherwise} \end{cases}. \tag{12}$$

The computation of the log-likelihood value for any given topology can be huge due to the large size of the transition matrices with  $2^{2d}$  elements. The computational complexity of log-likelihood evaluation is of order  $O(2^{2d}T)$ , where  $T$  is the time horizon. This substantial computational burden (especially in systems with a large number of components) is the motivation to come up with more efficient ways to solve the problem presented in Eq. 4.

#### 3.2 Bayesian optimization for topology optimization

This article proposes a Bayesian optimization approach for scalable topology inference of regulatory networks observed through temporally sparse data. Bayesian optimization (BO) (Frazier, 2018) is a well-known approach that has been extensively used in recent years for optimization problems in domains with expensive to evaluate objective functions. BO has shown great promise in increasing the automation and the quality of the optimization tasks (Shahriari et al., 2016). In this paper, we are dealing with an expensive-to-evaluate likelihood function. A major issue in employing the conventional BO is its ability of dealing with continuous search spaces, whereas the search space in our problem is the topology of regulatory networks, which takes a large combinatorial space. Therefore, some key changes need to be applied to the original BO formulation so that it can be adapted to our problem. The main concepts of this approach are explained in detail in the following paragraphs.

### 3.2.1 GP Model over the Log-Likelihood Function

The transition matrix ( $M^\theta$ ) in Eq. 7 makes the log-likelihood function evaluation in Eqs. 4, 11 computationally expensive, especially when dealing with large scale regulatory networks. Therefore, it is vital to come up with an efficient way of searching over the topology space. In this article, the log-likelihood function  $L(\cdot)$  is modeled using the Gaussian Process (GP) regression. The GP (Rasmussen and Williams, 2006) is mostly defined over continuous spaces, primarily due to the possibility of defining kernel functions that model the correlation over continuous spaces. In our case, the parameters are discrete interactions (i.e., parameters of the connectivity matrix that take +1, 0, or -1), which prevent constructing the GP model for representing the log-likelihood function over topology space.

This paper takes advantage of the topology structure of GRNs, encoded in connectivity matrix in (3), and defines the following GP model:

$$\mathcal{L}(\theta) = \mathcal{GP}(\mu(\theta), k(\theta, \theta)), \tag{13}$$

where  $\mu(\cdot)$  shows the mean function, and  $k(\cdot, \cdot)$  indicates the topology-inspired kernel function. The mean function,  $\mu(\cdot)$ , in Eq. 13 represents the prior shape of the log-likelihood function over all the topologies. One possible choice for the mean function is the constant mean function. This mean function carries a single hyperparameter, which can be learned along with the kernel hyperparameters.

Knowing that each parameter vector  $\theta$  corresponds to a connectivity matrix  $C^\theta$ , the structurally-inspired kernel function is defined as:

$$k(\theta, \theta') = \sigma_f^2 \exp\left(-\frac{\|C^\theta - C^{\theta'}\|^2}{l}\right), \tag{14}$$

where  $\|V\|^2$  is the sum of squares of elements of  $V$ ,  $C^\theta$  and  $C^{\theta'}$  represent the connectivity matrices related to topologies  $\theta$  and  $\theta'$  respectively,  $l$  is the length-scale, and  $\sigma_f^2$  is the scale factor hyperparameters. These hyperparameters quantify how close the topologies are to each other. The more similar two topologies are (i.e., less difference in the connectivity matrices), the more they are correlated, and the kernel function value will be higher for them. While, for more distinct topologies, the kernel will have smaller values.

Figure 2 represents an example of a few possible topologies for a GRN with two genes. These four possible topologies differ in one or two interactions. If the log-likelihood value for topology  $\theta^1$  is calculated, this information can be used for predicting log-likelihood values for other topologies. The connectivity matrices for these topologies can be expressed as:

$$C^{\theta^1} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, C^{\theta^2} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, C^{\theta^3} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, C^{\theta^4} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \tag{15}$$

The correlation between topology  $\theta^1$  and all the aforementioned topologies,  $\Theta^{\text{sub}} = \{\theta^1, \theta^2, \theta^3, \theta^4\}$  are calculated based on Eq. 14, and expressed through the following kernel vector:

$$\mathbf{K}_{(\theta^1, \Theta^{\text{sub}})} = \begin{bmatrix} k(\theta^1, \theta^1) & k(\theta^1, \theta^2) & k(\theta^1, \theta^3) & k(\theta^1, \theta^4) \\ \sigma_f^2 & \frac{\sigma_f^2}{\exp(1)} & \frac{\sigma_f^2}{\exp(4)} & \frac{\sigma_f^2}{\exp(16)} \end{bmatrix}, \tag{16}$$

where the length-scale hyperparameter is assumed to be 1. It can be seen that topology  $\theta^1$  has the maximum correlation with itself, and the correlation rate decreases when we move from topology  $\theta^1$  to  $\theta^4$ . This can also be understood in terms of the differences between the interacting parameters, expressed in the connectivity matrices in Eq. 15. Topology  $\theta^2$  is different from  $\theta^1$  in only missing interaction from gene 2 to gene 1. This results in a correlation of  $k(\theta^1, \theta^2) = \frac{\sigma_f^2}{\exp(1)}$  between these two topologies. Further, in Eq. 15 we can see that the interaction from gene 2 to gene 1 in model  $\theta^1$  is suppressive (-1), whereas the same interaction is activating (+1) in model  $\theta^3$ . This leads to smaller correlation between topologies  $\theta^1$  and  $\theta^3$ ,  $k(\theta^1, \theta^3) = \frac{\sigma_f^2}{\exp(4)}$ , in comparison to the correlation between topologies  $\theta^1$  and  $\theta^2$ . Finally, in Eq. 15, it can be seen that  $\theta^1$  and  $\theta^4$  have two opposite types of interactions, leading to the  $k(\theta^1, \theta^4) = \frac{\sigma_f^2}{\exp(16)}$ , which is the smallest correlation between  $\theta^1$  and all the other topologies.

The GP model has the capability of providing the Bayesian representation of the likelihood function across the topology space. Let  $\theta_{1:t} = (\theta_1, \dots, \theta_t)$  be the first  $t$  samples from the parameter space (i.e., samples from the topology candidates) with the associated log-likelihood values  $L_{1:t} = [L_1, \dots, L_t]^T$  (i.e.,  $L_1 = L(\theta_1)$  in Eq. 11). The posterior distribution of  $\mathcal{L}(\theta)$  in Eq. 13 is derived as:

$$\mathcal{L}(\theta) | \theta_{1:t}, L_{1:t} \sim \mathcal{N}(\mu_\theta^t, \Sigma_\theta^t), \tag{17}$$

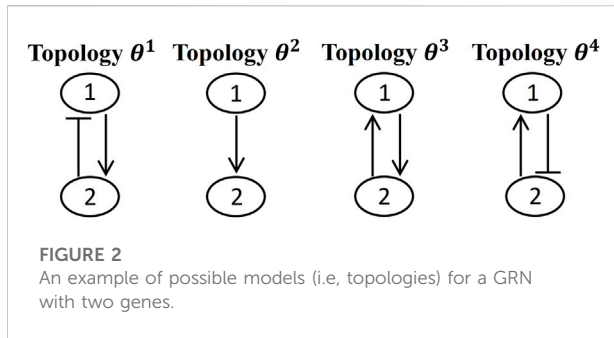
where  $\mu_\theta^t$  and  $\Sigma_\theta^t$  are the mean and variance for a specific model  $\theta \in \Theta$  respectively. These values can be obtained as:

$$\begin{aligned} \mu_\theta^t &= \mu(\theta) + \mathbf{K}_{(\theta, \theta_{1:t})} \mathbf{K}_{(\theta_{1:t}, \theta_{1:t})}^{-1} (L_{1:t} - \boldsymbol{\mu}(\theta_{1:t})), \\ \Sigma_\theta^t &= \mathbf{k}_{(\theta, \theta)} - \mathbf{K}_{(\theta, \theta_{1:t})} \mathbf{K}_{(\theta_{1:t}, \theta_{1:t})}^{-1} \mathbf{K}_{(\theta, \theta_{1:t})}^T, \end{aligned} \tag{18}$$

where  $\boldsymbol{\mu}(\theta_{1:t}) = [\mu(\theta_1), \dots, \mu(\theta_t)]^T$ , and

$$\mathbf{K}_{(\Theta, \Theta')} = \begin{bmatrix} k(\theta_1, \theta'_1) & \dots & k(\theta_1, \theta'_r) \\ \vdots & \ddots & \vdots \\ k(\theta_r, \theta'_1) & \dots & k(\theta_r, \theta'_r) \end{bmatrix}, \tag{19}$$

for  $\Theta = \{\theta_1, \dots, \theta_r\}$ ,  $\Theta' = \{\theta'_1, \dots, \theta'_r\}$ . Using the aforementioned formulation, the GP constructs the log-likelihood function as a zero-mean Bayesian surrogate model with covariance  $k(\cdot, \cdot)$ . Further, at iteration  $t$ , the log-likelihood function can be computed by employing the already chosen and evaluated log-likelihood values for topologies  $\theta_{1:t}$ , i.e.,  $L_{1:t}$ . The uncertainty of the



surrogate model will be reduced as we evaluate the likelihood function for more topologies.

The GP hyperparameters, which consist of the hyperparameters of the topology-inspired kernel function and the mean function, can be learned by optimizing the marginal likelihood function of the GP model at each iteration through:

$$(L_{1:t} | \theta_{1:t}) \sim \mathcal{N}(\mu(\theta_{1:t}), \mathbf{K}_{(\theta_{1:t}, \theta_{1:t})}). \quad (20)$$

### 3.2.2 Sequential Topology Optimization

The notion of efficient topology optimization is to come up with an efficient way of searching over all the topology space so that we utilize a minimum number of computationally expensive likelihood evaluations and eventually find the optimal topology, which yields the largest likelihood value.

As mentioned in Section 3.1, evaluation of the log-likelihood function for each topology is a computationally expensive task. Therefore, in here the sample-efficient and sequential topology selection is achieved as:

$$\theta_{t+1} = \underset{\theta \in \Theta}{\operatorname{argmax}} \alpha_t(\theta), \quad (21)$$

where  $\alpha_t(\theta)$  represents the *acquisition function* in the Bayesian optimization context, which is determined over the GP model posterior at iteration  $t$ . Multiple acquisition functions exist in the context of Bayesian optimization. For instance, *probability improvement* (Shahriari et al., 2016) is one of the most traditional acquisition functions, which makes selections to increase the likelihood of improvement in each iteration of BO. Other examples for acquisition functions include *expected improvement* (Mockus et al., 1978; Jones et al., 1998; Brochu et al., 2010), *upper confidence bound* (Auer, 2003), *knowledge gradient* (Wu et al., 2017; Frazier, 2009), and *predictive entropy search* (Henrandez-Lobato et al., 2014). In this work, we use the expected improvement acquisition function, which is the most commonly used acquisition function. This acquisition function balances the exploration and exploitation trade-off, and furthermore has a closed form solution. The expected improvement acquisition function is defined as (Mockus et al., 1978; Jones et al., 1998):

$$\alpha_t(\theta) = (\mu_\theta^t - L_{\max}^t) \Phi\left(\frac{\mu_\theta^t - L_{\max}^t}{\sqrt{\Sigma_\theta^t}}\right) + \sqrt{\Sigma_\theta^t} \phi\left(\frac{\mu_\theta^t - L_{\max}^t}{\sqrt{\Sigma_\theta^t}}\right), \quad (22)$$

Where  $\phi(\cdot)$  and  $\Phi(\cdot)$  refer to the probability density function and cumulative density function of standard normal distribution,  $L_{\max}^t = \max\{L_1, \dots, L_t\}$  is the maximum log-likelihood value until the latest turn, and  $\mu_\theta^t$  and  $\Sigma_\theta^t$  are the mean and variance of the GP model at iteration  $t$  as defined in Eq. 18.

The acquisition function in Eq. 22 holds a closed-form solution and requires the mean and variance of the GP model for any given topology. To solve Eq. 21 for large regulatory networks with a large number of unknown interactions, we can implement some heuristic optimization methods including particle swarm optimization technique (Kennedy and Eberhart, 1995), genetic algorithm (Anderson and Ferris, 1994; Whitley, 1994), or the breadth-first local search (BFLS) (Atabakhsh, 1991) to obtain the model with the largest acquisition value. After the model with maximum acquisition value ( $\theta_{t+1}$ ) is chosen, the next log-likelihood evaluation is carried for topology  $\theta_{t+1}$  to derive the log-likelihood value  $L_{t+1}$ . The GP model is then updated based on all the new information, defined as  $\theta_{1:t+1} = (\theta_{1:t}, \theta_{t+1})$  and  $L_{1:t+1} = [L_{1:t}, L_{t+1}]^T$ .

The proposed Bayesian topology optimization continues its sequential process over all the topology space of the regulatory networks for a fixed number of turns, or until no significant change in the maximum log-likelihood value in consecutive iterations is spotted. When the optimization ends, the topology with the largest evaluated likelihood value is selected as the system topology, meaning that:

$$\theta^* := \theta_{i^*}, \text{ where } i^* = \underset{i=1, \dots, t+1}{\operatorname{argmax}} L_i. \quad (23)$$

The inference process consists of three main components. Figure 3 represents the schematic diagram of the proposed method. The GP model predicts the log-likelihood values over the possible topology candidates, denoted by the black dots in Figure 3. The red dots denote the evaluated log-likelihood values for the selected topologies up to the current iteration. Using the posterior distribution of the GP model, the next topology with the highest acquisition function is selected, followed by the log-likelihood evaluation for the selected topology. The GP is then updated based on the selected topology and the evaluated log-likelihood, and this sequential process continues until a stopping criterion is met.

The detailed steps of the proposed inference method are described in Algorithm 1.  $\Theta$  denotes the topology space, and  $D_{1:T}$  represents the available data. Line 3 to line 8 of the algorithm creates the state index associated with the data  $D_{1:T}$ . The sequential topology optimization process is then carried out from line 10 to line 19, where in each loop, the log-likelihood value for a topology selected by the proposed

---

```

1: Topology space  $\Theta$ ; data  $D_{1:T}$ ; initialize the hyperparameters of the Gaussian process  $\mathcal{GP}(\mu(\cdot), k(\cdot, \cdot))$ ,
    $t = 0$ .
2: Arbitrary enumeration of the possible Boolean state vectors:  $\mathbf{X} = [\mathbf{x}^1, \dots, \mathbf{x}^{2^d}]$ 
3: Initialization:  $I = \vec{0}_{1 \times T}$ 
4: for  $k = 1, 2, \dots, T$  do
5:   for  $i = 1, \dots, 2^d$  do
6:      $I_k = \begin{cases} i & \text{if } D_k \text{ is not missing and } D_k = \mathbf{X}_i \\ 0 & \text{if } D_k \text{ is missing} \end{cases}$ 
7:   end for
8: end for
9: repeat
   Sequential Topology Selection
10:  Pick the topology with maximum acquisition value:  $\theta_{t+1} = \operatorname{argmax}_{\theta \in \Theta} \alpha_t(\theta)$  — Eq. 21
   Log-Likelihood Computation
11:  Initialization:  $\Pi_{0|0}^{\theta_{t+1}}(i) = P(\mathbf{X}_0 = \mathbf{x}^i \mid \theta_{t+1})$ , for  $i = 1, \dots, 2^d$ ,  $L_{t+1} = 0$ .
12:  for  $k = 1, 2, \dots, T$  do
13:    Predictive Posterior Distribution:  $\Pi_{k|k-1}^{\theta_{t+1}} = M^{\theta_{t+1}} \Pi_{k-1|k-1}^{\theta_{t+1}}$ .
14:    Posterior Distribution:  $\Pi_{k|k}^{\theta_{t+1}}(i) = \begin{cases} \Pi_{k|k-1}^{\theta_{t+1}}(i) & \text{if } I_k = 0, \\ 1 & \text{if } i = I_k, \\ 0 & \text{otherwise} \end{cases}$ 
15:    Log-Likelihood Update:  $L_{t+1} = L_{t+1} + \log P(I_k \mid I_{1:k-1}, \theta_{t+1})$ ,
      where  $P(I_k \mid I_{1:k-1}, \theta_{t+1}) = \begin{cases} \Pi_{k|k-1}^{\theta_{t+1}}(I_k) & \text{if } I_k \neq 0 \\ 1 & \text{otherwise} \end{cases}$ 
16:  end for
   Update the GP Model
17:  Update the hyperparameters of the GP according to  $(\theta_{1:t+1}, L_{1:t+1})$ .
18:   $t = t + 1$ .
19: until the stopping criterion is met
20: The Inferred Topology:  $\theta^* := \theta_{i^*}$ , where  $i^* = \operatorname{argmax}_{i=1, \dots, t} L_i$ .

```

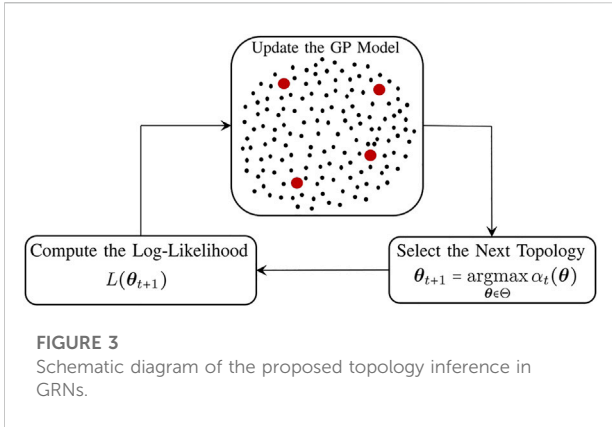
---

**Algorithm 1.** The Proposed method for inference of regulatory networks through temporally-sparse data.

Bayesian optimization technique is computed, followed by the GP posterior update and the next topology selection. Finally, upon the termination of the inference process, the topology with the maximum log-likelihood is chosen in line 20 as the inferred topology. The computation of the log-likelihood determines the complexity of the algorithm at each step of our proposed method, which is of order  $O(2^{2^d T})$ . This means that the complexity at each step of the proposed method is the same as one log-likelihood evaluation. The log-likelihood evaluation at each iteration is used to update our knowledge (posterior), and to help choose the best candidate for future iterations.

## 4 Numerical experiments

The code repository for replicating the numerical experiments of this paper is included in the data availability statement at the end of this paper. The well-known mammalian cell-cycle network (Fauré et al., 2006) is used to evaluate the performance of our proposed method. Figure 4 presents the pathway diagram of this network. The state vector for this network is assumed as the following  $\mathbf{x} = (\text{CycD}, \text{Rb}, \text{p27}, \text{E2F}, \text{CycE}, \text{CycA}, \text{Cdc20}, \text{Cdh1}, \text{UbcH10}, \text{CycB})$ . The division of mammalian cells depends on the overall organism growth, controlled using signals that activate cyclin D (CycD) in the cell. As can be seen in the state vector, the mammalian cell-cycle network



**FIGURE 3**  
Schematic diagram of the proposed topology inference in GRNs.

contains 10 genes ( $d = 10$ ). The settings used for our experiments are as follows: data length of 100 ( $k = 100$ ), process noise of 0.1 ( $p = 0.1$ ), and missing data percentage (sparsity) of 50%. Furthermore, 10 regulations of the connectivity matrix are assumed to be unknown ( $L = 10$ ), and a maximum of 100 likelihood evaluations are used for the inference process. All the parameters used throughout the numerical experiments are expressed in Table 1.

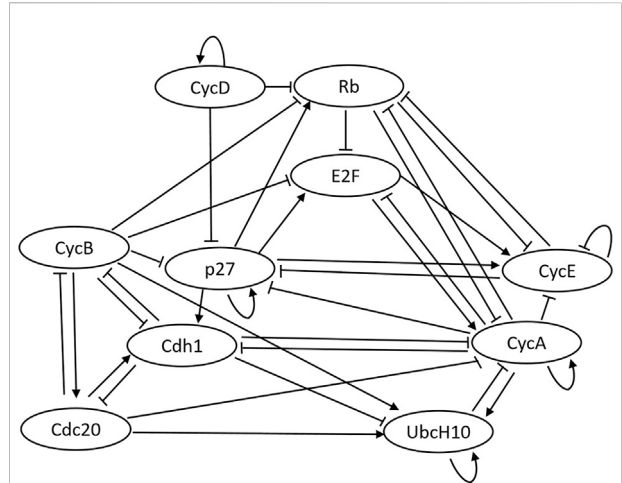
The connectivity matrix and bias vector in Eq. 3 for the mammalian cell-cycle network can be written as:

$$\mathbf{C} = \begin{bmatrix} +1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & +1 & 0 & -1 & -1 & 0 & 0 & 0 & -1 \\ -1 & 0 & +1 & 0 & -1 & -1 & 0 & 0 & 0 & -1 \\ 0 & -1 & +1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\ 0 & -1 & +1 & +1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & +1 & 0 & +1 & -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & +1 \\ 0 & 0 & +1 & 0 & 0 & -1 & +1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & +1 & +1 & -1 & +1 & +1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \end{bmatrix}, \quad (24)$$

$$\mathbf{b} = \left[ \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \quad \frac{-1}{2} \right]^T.$$

In this section, we are assuming that the connectivity matrix is not fully known. This network has 10 genes, and there is a total of  $2^{10} = 1,024$  possible states for this network. Consequently, the transition matrix size is  $2^{10} \times 2^{10}$ , which causes the likelihood evaluation to be computationally expensive for any possible topology. Using our proposed method, we show that the optimal topology with the largest log-likelihood value can be inferred with few likelihood evaluations; hence, we offer an efficient search over all possible topologies.

In all of the experiments, 10 unknown interactions ( $c_{ij}$ ) were considered. Each of the unknown interactions can take their values in the set  $\{+1, 0, -1\}$ , which leads to  $3^{10} = 59,049$  different possible system models, i.e.,  $\Theta = \{\theta^1, \dots, \theta^{3^{10}}\}$ . The 10 randomly chosen unknown regulations, which are elements of the connectivity matrix in Eq. 24, are:



**FIGURE 4**  
Pathway diagram for the cell-cycle network.

**TABLE 1** Parameter values of mammalian cell-cycle network experiments.

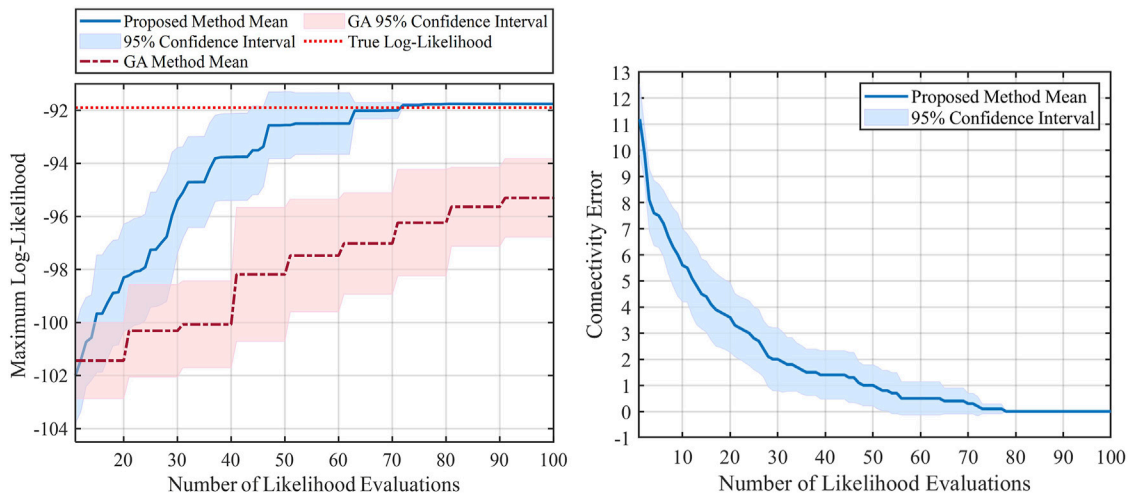
Parameter	Value
Trajectory Length, $k$	100
Number of Likelihood Evaluations	100
Number of Genes, $d$	10
Number of Unknown Regulations, $L$	10
Process Noise, $p$	0.1
Missing Data Percentage	50%

$$c_{21} = -1, \quad c_{35} = -1, \quad c_{310} = -1, \quad c_{42} = -1, \quad c_{54} = +1 \\
 c_{67} = -1, \quad c_{69} = -1, \quad c_{83} = +1, \quad c_{96} = +1, \quad c_{98} = -1. \quad (25)$$

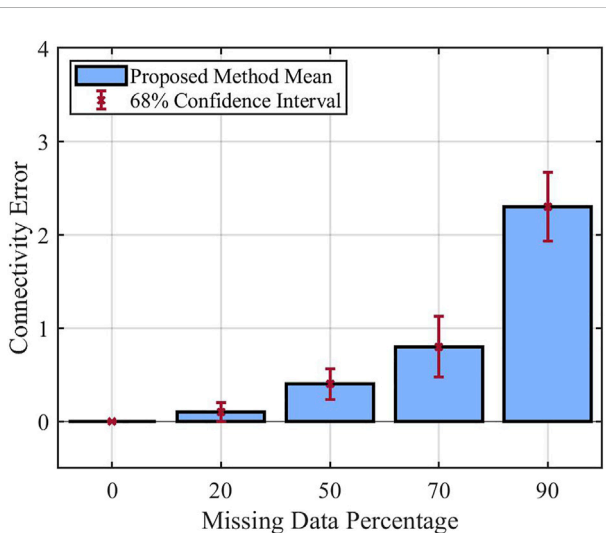
We also considered a uniform prior distribution for the initial states, i.e.,  $\Pi_{010}^\theta(i) = \frac{1}{2^{10}}$  for all  $\theta \in \Theta$  and  $i = 1, 2, \dots, 2^{10}$ . Furthermore, all the experiments are repeated for 10 independent runs, and the average results along the confidence bounds are reported in all the figures. Note that the randomness of early results come from the process noise ( $p$ ), and the way the sequential topology optimization is being performed in each run.

For the first set of experiments, the performance of the proposed method is shown using two plots in Figure 5. The left plot represents the progress of the log-likelihood value of the inferred model with respect to the number of likelihood evaluations, meaning that it shows the maximum log-likelihood value obtained during the optimization process. Larger log-likelihood values mean that the chosen model can better represent the true model (i.e., the available

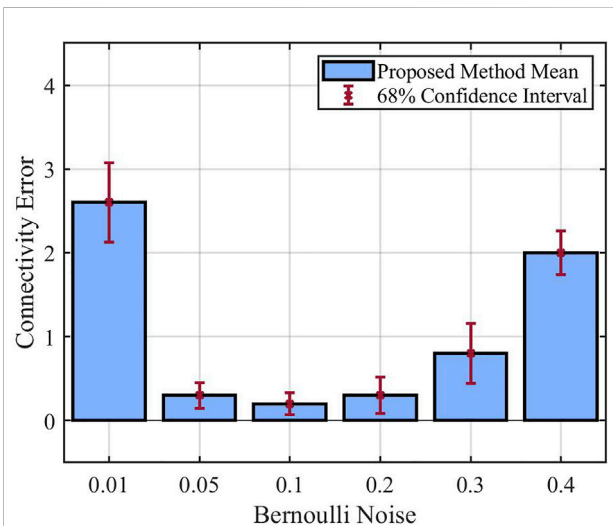




**FIGURE 5**  
Results of the mammalian cell-cycle network with 10 unknown interactions.



**FIGURE 6**  
Performance of the proposed method with respect to percentage of missing data.



**FIGURE 7**  
Performance of the proposed method in presence of different Bernoulli noise.

data is more likely to come from models with larger likelihood values). As a comparison, we also repeated the same experiment using Genetic Algorithm (GA) (Anderson and Ferris, 1994; Whitley, 1994), which is a powerful and well-known solver for non-continuous problems. By looking at the left plot in Figure 5, we can see that the inference by the proposed method, indicated by the solid blue line, is better than the GA method (dashed red line). This superiority can be seen in terms of the mean and confidence intervals in Figure 5. As we evaluate more likelihoods for different models, the likelihood of the proposed method's inferred model gets closer to the optimal log-likelihood value,

indicated by the dotted red line. Hence, our proposed method is capable of reaching a better log-likelihood with less number of likelihood evaluations and has a more efficient way of searching over all the possible models. Furthermore, the 95% confidence interval is illustrated in the same plot for both methods during this experiment. We can observe that the proposed method's confidence interval keeps getting smaller, and roughly after 70 evaluations, the confidence interval tends to go zero. This indicates the robustness of the proposed method, where after roughly 70 iterations, the log-likelihood gets to its optimal value at different independent runs. By contrast, the results from the GA

still show a large confidence interval even after 100 evaluations, and its average is far less than the optimal log-likelihood value.

The right plot of Figure 5 shows the progress of the connectivity error during the optimization process (i.e., number of likelihood evaluations) obtained by the proposed method. Let  $C^*$  be the vectorized true connectivity matrix indicated in Eq. 24, and  $C^t$  be the vectorized inferred connectivity matrix at  $t$ th likelihood evaluation. The connectivity error at iteration  $t$  is defined as  $\|C^* - C^t\|_1$ . Evidently, we will have a better estimate of the true model as this error gets closer to zero. In the right plot, we can see that the connectivity error decreases as we do more evaluations, and after about 75 likelihood evaluations, the error gets to zero, meaning that we successfully inferred the true connectivity matrix. Also, as expected, we can see that the 95% confidence interval gets smaller as we do more evaluations and eventually gets close to zero after about 75 evaluations.

In the second set of experiments, we aim to investigate the effect of missing data percentage on the performance of the proposed method. It is expected that with more missing data, it would be more difficult to infer the relationship between different components of the system; hence the connectivity error for the inferred model would be larger. For these experiments, we changed the missing data percentage from 0% to 90% and used Bernoulli noise value 0.2. Other parameters are fixed based on Table 1. The mean of the inferred models' connectivity error obtained from these experiments, along with their 68% confidence interval are presented as bar plots in Figure 6. As expected, these results demonstrate that the mean of connectivity error increases as the missing data percentage gets larger.

The final set of experiments focuses on how the Bernoulli noise affects the performance of the proposed method. In all these experiments, we consider 50% missing data percentage, and we change the Bernoulli noise from 0.01 to 0.4. For performance comparison, the mean of the inferred models' connectivity error derived from these experiments is shown using bar plots in Figure 7. In this bar plot, we can observe that the connectivity error is large for the Bernoulli noise of 0.01. As the noise increases to 0.05 and 0.1, the connectivity error keeps decreasing. However, increasing the noise to 0.2, 0.3, and finally 0.4 results in a continuous increase in the connectivity error. These results demonstrate the relationship between network stochasticity and data informativity needed for the inference process. For a small process noise ( $p = 0.01$ ), the network is typically trapped in attractor states, which precludes the observation of the entire state space. This leads to the issue of statistical non-identifiability, which refers to the situation where multiple models are not clearly distinguishable using the available data. Once the noise value is slightly increased ( $p = 0.05$ ,  $p = 0.1$ ), the network gets out of its attractor states more often, which enhances the performance of the inference process. Finally, for too large process noise values ( $p = 0.2$ ,  $p = 0.3$ ,  $p = 0.4$ ), the state transitions become more chaotic, making it more difficult to infer the true relations between the components.

## 5 Conclusion

This paper presents a highly scalable topology inference method for gene regulatory networks (GRNs) observed through temporally sparse data. The Boolean network model is used for capturing the dynamics of the GRNs. The inference process consists of inferring the interactions between genes or equivalently selecting a topology for the system among all the possible topologies that have the highest likelihood value. Evaluating the likelihood function for any given topology is expensive, preventing exhaustive search over the large possible topology space. The proposed method models the log-likelihood function by a Gaussian Process (GP) model with a structurally-inspired kernel function. This GP model captures the correlation between different possible topologies and provides the Bayesian representation of the log-likelihood function. Using the posterior distribution of the GP model, Bayesian optimization is used to efficiently search over the topology space.

The high performance of our proposed method is shown using multiple experiments on the well-known mammalian cell-cycle network. We have also repeated all the experiments multiple times to obtain a confidence interval and further demonstrate the accuracy and robustness of the solutions obtained by our method. In the first experiment, we considered the topology inference of the mammalian cell-cycle network with 10 unknown interactions and 50% missing data. From comparing the results of topology inference using our proposed method and genetic algorithm, we observed that our method is more efficient in searching over the topology space and reaches an optimal model with fewer likelihood evaluations. Meanwhile, the small confidence interval of our method justified the robustness of the solutions. The second experiment investigated the effect of missing data on the performance of the proposed inference method. From the results, we understand that as expected, with more missing data, the method's accuracy reduces, and the inference error becomes larger. Finally, in the third experiment, we studied the performance of our method in the presence of different Bernoulli noise (i.e., stochasticity in the state process). The results show that for small stochasticity, the accuracy of the inference is low, as the system spends most of its time in a few states (i.e., attractors) and the interactions between different components of the system are not distinguishable. As the stochasticity increases, the accuracy of the proposed method increases (as the error decreases) until a certain point, and after that again, the accuracy starts decreasing. This is because too much stochasticity turns the system into a more chaotic form, making the inference of the true model more challenging.

## Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: <https://github.com/ImaniLab/Frontiers-2022>.

## Author contributions

MA and MI contributed to conceptualization, methodology, validation, formal analysis, and investigation of the study. MA was responsible for coding, resources, and data curation of the work. Supervision and project administration was done by MI. Both authors contributed to writing and editing of the manuscript, and they have read and agreed to the submitted version of the manuscript.

## Funding

This work has been supported in part by the National Institutes of Health award 1R21EB032480-01, National Science Foundation award IIS-2202395, ARMY Research Office award W911NF2110299, and Oracle Cloud credits

## References

- Abou-Jaoudé, W., Traynard, P., Monteiro, P., Saez-Rodriguez, J., Helikar, T., Thieffry, D., et al. (2016). Logical modeling and dynamical analysis of cellular networks. *Front. Genet.* 7, 94. doi:10.3389/fgene.2016.00094
- Anderson, E. J., and Ferris, M. C. (1994). Genetic algorithms for combinatorial optimization: The assemble line balancing problem. *ORSA J. Comput.* 6, 161–173. doi:10.1287/ijoc.6.2.161
- Atabakhsh, H. (1991). A survey of constraint based scheduling systems using an artificial intelligence approach. *Artif. Intell. Eng.* 6, 58–73. doi:10.1016/0954-1810(91)90001-5
- Auer, P. (2003). Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.* 3, 397–422.
- Barman, S., and Kwon, Y.-K. (2017). A novel mutual information-based boolean network inference method from time-series gene expression data. *PLoS one* 12, e0171097. doi:10.1371/journal.pone.0171097
- Barman, S., and Kwon, Y.-K. (2018). A boolean network inference from time-series gene expression data using a genetic algorithm. *Bioinformatics* 34, i927–i933. doi:10.1093/bioinformatics/bty584
- Brochu, E., Cora, V. M., and de Freitas, N. (2010). A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arxiv. doi:10.48550/ARXIV.1012.2599
- Cheng, D., and Zhao, Y. (2011). Identification of boolean control networks. *Automatica* 47, 702–710. doi:10.1016/j.automatica.2011.01.083
- Dougherty, E., and Qian, X. (2013). Validation of gene regulatory network inference based on controllability. *Front. Genet.* 4, 272. doi:10.3389/fgene.2013.00272
- Fauré, A., Naldi, A., Chaouiya, C., and Thieffry, D. (2006). Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle. *Bioinformatics* 22, e124–e131. doi:10.1093/bioinformatics/btl210
- Frazier, P. (2009). Knowledge-gradient methods for statistical learning. Princeton, NJ: Department of Operations Research and Financial Engineering, Princeton University. Dissertation.
- Frazier, P. I. (2018). A tutorial on bayesian optimization. arxiv. doi:10.48550/ARXIV.1807.02811
- Godsill, S. J., Doucet, A., and West, M. (2004). Monte Carlo smoothing for nonlinear time series. *J. Am. Stat. Assoc.* 99, 156–168. doi:10.1198/016214504000000151
- Hajiramezani, E., Imani, M., Braga-Neto, U., Qian, X., and Dougherty, E. R. (2019). Scalable optimal Bayesian classification of single-cell trajectories under regulatory model uncertainty. *BMC genomics* 20, 435. doi:10.1186/s12864-019-5720-3
- Hashimoto, R. F., Kim, S., Shmulevich, I., Zhang, W., Bittner, M. L., and Dougherty, E. R. (2004). Growing genetic regulatory networks from seed genes. *Bioinformatics* 20, 1241–1247. doi:10.1093/bioinformatics/bth074
- Henrández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. (2014). “Predictive entropy search for efficient global optimization of black-box functions,” in Proceedings of the 27th International Conference on Neural Information Processing Systems, 918–926.
- Hürzeler, M., and Künsch, H. R. (1998). Monte Carlo approximations for general state-space models. *J. Comput. Graph. Statistics* 7, 175–193. doi:10.1080/10618600.1998.10474769
- Imani, M., and Braga-Neto, U. M. (2017). Maximum-likelihood adaptive filter for partially observed Boolean dynamical systems. *IEEE Trans. Signal Process.* 65, 359–371. doi:10.1109/tsp.2016.2614798
- Imani, M., and Ghoreishi, S. F. (2021). Two-stage Bayesian optimization for scalable inference in state space models. *IEEE Trans. Neural Netw. Learn. Syst.* 33, 5138–5149. doi:10.1109/tnnls.2021.3069172
- Imani, M., Dehghanasari, R., Braga-Neto, U. M., and Dougherty, E. R. (2018). Sequential experimental design for optimal structural intervention in gene regulatory networks based on the mean objective cost of uncertainty. *Cancer Inf.* 17, 117693511879024. doi:10.1177/1176935118790247
- Imani, M., Ghoreishi, S. F., Allaire, D., and Braga-Neto, U. (2019). MFBO-SSM: Multi-fidelity Bayesian optimization for fast inference in state-space models. *Proc. AAAI Conf. Artif. Intell.* 33, 7858–7865. doi:10.1609/aaai.v33i01.33017858
- Imani, M., Dougherty, E., and Braga-Neto, U. (2020). Boolean Kalman filter and smoother under model uncertainty. *Automatica* 111, 108609. doi:10.1016/j.automatica.2019.108609
- Johansen, A. M., Doucet, A., and Davy, M. (2008). Particle methods for maximum likelihood estimation in latent variable models. *Stat. Comput.* 18, 47–57. doi:10.1007/s11222-007-9037-8
- Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *J. Glob. Optim.* 13, 455–492. doi:10.1023/a:1008306431147
- Kantas, N., Doucet, A., Singh, S. S., Maciejowski, J., and Chopin, N. (2015). On particle methods for parameter estimation in state-space models. *Stat. Sci.* 30, 328–351. doi:10.1214/14-sts511
- Karbalayghareh, A., Braga-neto, U., and Dougherty, E. R. (2018). Intrinsically bayesian robust classifier for single-cell gene expression trajectories in gene regulatory networks. *BMC Syst. Biol.* 12, 23–10. doi:10.1186/s12918-018-0549-y
- Kennedy, J., and Eberhart, R. (1995). “Particle swarm optimization,” in Proceedings of ICNN’95-International Conference on Neural Networks (IEEE), 1942–1948.
- Lähdesmäki, H., Shmulevich, I., and Yli-Harja, O. (2003). On learning gene regulatory networks under the boolean network model. *Mach. Learn.* 52, 147–167. doi:10.1023/A:1023905711304
- Marshall, S., Yu, L., Xiao, Y., and Dougherty, E. R. (2007). Inference of a probabilistic boolean network from a single observed temporal sequence. *Eurasip J. Bioinforma. Syst. Biol.* 2007, 1–15. doi:10.1155/2007/32454

and related resources provided by the Oracle for Research program.

## Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Publisher’s note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- Mockus, J., Tiesis, V., and Zilinskas, A. (1978). The application of Bayesian methods for seeking the extremum. *Towards Glob. Optim.* 2, 2.
- Pusnik, Z., Mraz, M., Zimic, N., and Moskon, M. (2022). Review and assessment of boolean approaches for inference of gene regulatory networks. *Heliyon* 8, e10222. doi:10.1016/j.heliyon.2022.e10222
- Rasmussen, C. E., and Williams, C. (2006). *Gaussian processes for machine learning*. Cambridge, United States: MIT Press.
- Saadatpour, A., and Albert, R. (2013). Boolean modeling of biological regulatory networks: A methodology tutorial. *Methods* 62, 3–12. doi:10.1016/j.ymeth.2012.10.012
- Schön, T. B., Wills, A., and Ninness, B. (2011). System identification of nonlinear state-space models. *Automatica* 47, 39–49. doi:10.1016/j.automatica.2010.10.013
- Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and de Freitas, N. (2016). Taking the human out of the loop: A review of bayesian optimization. *Proc. IEEE* 104, 148–175. doi:10.1109/jproc.2015.2494218
- Shmulevich, I., and Dougherty, E. R. (2010). *Probabilistic boolean networks: The modeling and control of gene regulatory networks*. Philadelphia, United States: siam.
- Shmulevich, I., Dougherty, E. R., and Zhang, W. (2002). From Boolean to probabilistic Boolean networks as models of genetic regulatory networks. *Proc. IEEE* 90, 1778–1792. doi:10.1109/jproc.2002.804686
- Tan, Y., Neto, F. L., and Braga-Neto, U. (2020). Pallas: Penalized maximum likelihood and particle swarms for inference of gene regulatory networks from time series data. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 19, 1807–1816. doi:10.1109/tcbb.2020.3037090
- Whitley, D. (1994). A genetic algorithm tutorial. *Stat. Comput.* 4, 65–85. doi:10.1007/bf00175354
- Wills, A., Schön, T. B., Ljung, L., and Ninness, B. (2013). Identification of hammerstein-wiener models. *Automatica* 49, 70–81. doi:10.1016/j.automatica.2012.09.018
- Wu, J., Poloczek, M., Wilson, A. G., and Frazier, P. (2017). “Bayesian optimization with gradients” in *Advances in Neural Information Processing Systems* (Curran Associates, Inc.) 30.
- Wynn, M. L., Consul, N., Merajver, S. D., and Schnell, S. (2012). Logic-based models in systems biology: A predictive and parameter-free network analysis method. *Integr. Biol.* 4, 1323. doi:10.1039/c2ib20193c