Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 58-15 (2024) 1–6

Kernel-Based Particle Filtering for Scalable Inference in Partially Observed Boolean Dynamical Systems ⋆

Mohammad Alali * **Mahdi Imani** *

* *Northeastern University, Department of Electrical and Computer Engineering, Boston, MA, USA (e-mails: alali.m@northeastern.edu, m.imani@northeastern.edu).*

Abstract: This paper addresses the inference challenges associated with a class of hidden Markov models with binary state variables, known as partially observed Boolean dynamical systems (POBDS). POBDS have demonstrated remarkable success in modeling the ON and OFF dynamics of genes, microbes, and bacteria in systems biology, as well as in network security to represent the propagation of attacks among interconnected elements. Despite existing optimal and approximate inference solutions for POBDS, scalability remains a significant issue due to the computational cost associated with likelihood evaluations and the exploration of extensive parameter spaces. To overcome these challenges, this paper proposes a kernel-based particle filtering approach for large-scale inference of POBDS. Our method employs a Gaussian process (GP) to efficiently represent the expensive-to-evaluate likelihood function across the parameter space. The likelihood evaluation is approximated using a particle filtering technique, enabling the GP to account for various sources of uncertainty, including limited likelihood evaluations. Leveraging the GP's predictive behavior, a Bayesian optimization strategy is derived for effectively seeking parameters yielding the highest likelihood, minimizing the overall computational burden while balancing exploration and exploitation. The proposed method's performance is demonstrated using two biological networks: the mammalian cell-cycle network and the T-cell large granular lymphocyte leukemia network.

Keywords: Inference, Bayesian Optimization, Particle Filters, Partially-Observed Boolean Dynamical Systems, Biological Networks.

1. INTRODUCTION

Hidden Markov Models (HMMs) with binary state variables are a specialized and crucial class of models designed to capture the dynamic behaviors of systems or processes consisting of interconnected elements. This class, known as partially observed Boolean dynamical systems (POBDS) [Imani and Braga-Neto (2017); Imani et al. (2019)], has demonstrated remarkable success across diverse domains. Examples include biological networks in genomics and metagenomics, where the ON and OFF behavior of genes, microbes, and bacteria is effectively captured by the POBDS model [Puvsnik et al. (2022)]. Another widespread application is in network security, where the POBDS is utilized to capture the attack status and probabilistic propagation of attacks among interconnected elements [Kazeminajafabadi and Imani (2023)]. The POBDS can also be seen as a generalization of existing Boolean network models [Zhang (2023)], considering general-form stochasticity in the state process and partial observability of binary state variables through arbitrary (e.g., non-binary) observations.

A major objective is to construct the POBDS model according to noisy available observations. Several methods have been developed for the inference of POBDS, including optimal joint state and maximum likelihood (ML) inference [Imani and Braga-Neto (2017)], optimal Bayesian estimation and inference [Imani et al. (2019)], and particle filtering for scalable estimation and ML inference [Imani and Braga-Neto (2018)]. Despite their success, these approaches face challenges when dealing with systems with substantial uncertainty, typically modeled through several unknown parameters reflected in large parameter spaces. The computational cost associated with likelihood or posterior evaluations often prevents the inference of POMDP models, particularly when dealing with parameter spaces with a large or infinite number of selections. Additionally, finding the gradients of the likelihood function can be impossible (for discrete or mixed discrete and continuous parameter spaces), expensive (growing exponentially with the size of systems), and inaccurate (when the likelihood function is approximated).

To address the limitations of existing methods, this paper introduces a kernel-based particle filtering approach that represents the expansive likelihood function defined over a large parameter space using a surrogate model. The surrogate model is Gaussian process (GP) regression [Rasmussen and et al. (2006)], a non-parametric approach with great predictive capabilities according to the limited likelihood evaluations. Given the scale of the system, an

⋆ The authors acknowledge the support of the National Institute of Health award 1R21EB032480-01, National Science Foundation awards IIS-2311969 and IIS-2202395, ARMY Research Laboratory award W911NF2320179, ARMY Research Office award W911NF2110299, and Office of Naval Research award N00014-23-1-2850.

10.1016/j.ifacol.2024.08.495

auxiliary particle filter is employed for an efficient approximation of the likelihood function, where the approximation is factored into the construction of the GP model. The search for the model/parameter with the highest likelihood value, i.e., the ML inference solution, is obtained by developing a Bayesian optimization method. The Bayesian optimization performs a sequential search across the space of parameters based on the GP model's latest posterior, where a balance between exploration and exploitation is achieved to ensure the scalability of the inference over the large parameter spaces. The proposed inference approach simultaneously estimates the underlying state based on the performed auxiliary particle filter. The estimated state yields an approximate minimum mean squared error (MMSE) optimality corresponding to the ML estimate of the parameters. The efficacy of the proposed method is evaluated using the inference of gene regulatory networks observed through high-throughput sequencing data. Two well-known biological networks, the mammalian Cell-Cycle network [Alali and Imani (2024); Davidich and Bornholdt (2008)] and the T-cell Large Granular Lymphocyte (T-LGL) Leukemia network [Saadatpour et al. (2011)], are considered for performance analysis. The results in terms of the inference and state estimation accuracy, as well as the robustness of inference with respect to the system stochasticity level, are reported.

2. BACKGROUND AND PROBLEM FORMULATION

2.1 POBDS Model

Assume a network consisting of n components, where the state value of each component can be expressed as 0 or 1 (i.e., a gene status as OFF or ON, or a machine security as compromised or not compromised). The network state at time step k can be represented using a vector of size n consisting of the status of all components, i.e., $\mathbf{x}_k = [\mathbf{x}_k(1), ..., \mathbf{x}_k(n)]$, where $\mathbf{x}_k$ takes its value in one of the 2^n possible binary vectors , i.e., $\mathbf{x}_k \in \mathcal{X} = \{\mathbf{x}^1, ..., \mathbf{x}^{2^n}\}$. The connections/interactions among different components govern the state transitions, which are often probabilistic in practice. The stochasticity could be due to the unmodeled parts of the systems or the intrinsic stochasticity in the system behavior. The state process of POBDS is expressed through:

$$\mathbf{x}_k \sim P(. \mid \mathbf{x}_{k-1}, \boldsymbol{\theta}) \text{ (state process)}, \tag{1}$$

where $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$ are the states at time steps k and $k-1$, $P(\cdot)$ represents the probability mass function (discrete nature of state variables), and the parameter vector $\boldsymbol{\theta}$ represents the unknown parts of the state process. The dependency on the next system state to only the previous time step represents the Markovian assumption of the state process. $\boldsymbol{\theta} \in \Theta$ represents a realization of the state process, where Θ is a finite or infinite space depending on the problem structure.

The binary state variables $\mathbf{x}_{1:k}$ are often not directly observable but rather can be accessed through noisy observations $\mathbf{y}_{1:k}$. Following is the POBDS measurement process:

$$\mathbf{y}_k \sim P(. \mid \mathbf{x}_k, \boldsymbol{\theta}) \text{ (measurement process)}, \tag{2}$$

where the parameter vector $\boldsymbol{\theta}$ includes the unknown parts of the measurement process as well. Note that any POBDS can be described through the state and measurement processes introduced in (1) and (2).

2.2 Maximum Likelihood Estimation

In the context of the POBDS model, let $\boldsymbol{\theta} \in \Theta$ signify the parameters, with Θ denoting the parameter space. This parameter vector may include the parameters governing both the state and measurement processes detailed in (1) and (2). Let the available measurements be represented as $\mathbf{y}_{1:T} = (\mathbf{y}_1, ..., \mathbf{y}_T)$. The primary goal is to infer the parameters of the model based on these observed measurements. The Maximum Likelihood (ML) estimation of these parameters is defined as [Alali and Imani (2022); Ravari et al. (2024)]:

$$\hat{\boldsymbol{\theta}}^{\text{ML}} = \underset{\boldsymbol{\theta} \in \Theta}{\text{argmax}} \log P(\mathbf{y}_{1:T} \mid \boldsymbol{\theta}), \tag{3}$$

where $\log P(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})$ is the logarithm of the likelihood function, $P(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})$. This log-likelihood function can be expressed as:

$$L_T(\boldsymbol{\theta}) = \log P(\mathbf{y}_{1:T} \mid \boldsymbol{\theta}) = \sum_{k=1}^{T} \log P(\mathbf{y}_k \mid \mathbf{y}_{1:k-1}, \boldsymbol{\theta}). \tag{4}$$

The state estimation in the context of POBDS can be determined using the ML estimate of the parameters, represented as:

$$\hat{\mathbf{x}}_k^{\text{ML}} = \underset{\hat{\mathbf{x}}_k \in \{0,1\}^n}{\text{argmin}} \, \mathbb{E}\left[||\hat{\mathbf{x}}_k - \mathbf{x}_k||_2^2 \mid \mathbf{y}_{1:k}, \hat{\boldsymbol{\theta}}^{\text{ML}} \right], \tag{5}$$

for $k = 1, ..., T$. Here, $\hat{\mathbf{x}}_k$ denotes the estimated system state at time k, and $||.||_2^2$ is the squared L_2 norm of the vector. The set $\{1, 0\}^n$ encompasses all 2^n potential states. The ML state estimator, $\hat{\mathbf{x}}_{k|T}^{\text{ML}}$, as in (5), ensures minimal mean-square error (MMSE) optimality relative to the ML estimate of the model, $\hat{\boldsymbol{\theta}}^{\text{ML}}$.

According to the Boolean Kalman filter theorem [Imani and Braga-Neto (2017)], the solution for optimization in (5) can be achieved as:

$$\hat{\mathbf{x}}_k^{\text{ML}} = \overline{\mathbb{E}\left[\mathbf{x}_k \mid \mathbf{y}_{1:k}, \hat{\boldsymbol{\theta}}^{\text{ML}} \right]}, \tag{6}$$

where $\overline{\mathbf{v}}$ maps the elements greater than 0.5 to 1 and others to 0.

3. PROPOSED KERNEL-BASED PARTICLE FILTERING METHOD

This paper aims to allow scalable inference achieved through solving the optimization of $\text{argmax}_{\boldsymbol{\theta} \in \Theta} L_T(\boldsymbol{\theta})$, where $L_T(\boldsymbol{\theta}) = \log P(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})$. Two main challenges in solving this problem are: 1) a large and non-differentiable parameter space, which prevents going through all possible parameters or using gradient-based approaches to efficiently search the space; 2) the log-likelihood calculation for any given $\boldsymbol{\theta}$, i.e., $L_T(\boldsymbol{\theta})$, can be extremely expensive, which often occurs in large systems. In fact, the state space in POBDS grows exponentially with the number of elements in the systems, leading to exponentially larger costs for exact log-likelihood evaluations.

This paper tackles the aforementioned inference issues using the following two key elements: 1) incorporating the particle filtering method for a scalable approximation of the log-likelihood function; 2) effective search over large (and non-differentiable) parameter spaces using Bayesian optimization techniques. These two key elements, outlined below, ensure the scalability of inference with respect to the size of parameters and the systems, enabling inference for practical problems.

3.1 Particle Filtering Approximation of Log-Likelihood

The transition probability in POBDS refers to the likelihood of moving from one state to another. The state space of POBDS takes in a finite set of 2^n binary vectors for a system of n components. Therefore, The transition probabilities can be expressed using a matrix of $2^n \times 2^n$, which becomes computationally intractable in large systems. Particle filtering approaches have been extensively used as tractable solutions for state and parameter estimation in systems with large or continuous state spaces. This paper employs an auxiliary particle filtering approach (APF) [Pitt and Shephard (1999)] to enable efficient computation of the log-likelihood function.

Let the auxiliary variable, ζ_k, be an index that identifies the particles from the preceding step. For any $\boldsymbol{\theta} \in \Theta$, the APF represents the joint state and auxiliary variable as:

$$P(\mathbf{x}_k, \zeta_k | \mathbf{y}_{1:k}, \boldsymbol{\theta}) \propto P(\mathbf{x}_{k-1}, \zeta_k \mid \mathbf{y}_{1:k-1}, \boldsymbol{\theta})\, P(\mathbf{x}_k | \mathbf{x}_{k-1}, \zeta_k, \boldsymbol{\theta})\, P(\mathbf{y}_k | \mathbf{x}_k, \boldsymbol{\theta}), \tag{7}$$

for $\zeta_k = 1, \ldots, N$.

Let the posterior distribution at time step $k-1$ be approximated as:

$$P(\mathbf{x}_{k-1} = \mathbf{x}^l \mid \mathbf{y}_{1:k-1}, \boldsymbol{\theta}) \approx \frac{1}{N} \sum_{i=1}^{N} W^{\boldsymbol{\theta}}_{k-1,i} 1_{\mathbf{x}^l = \mathbf{x}^{\boldsymbol{\theta}}_{k-1,i}}, \tag{8}$$

for $l = 1, ..., 2^n$; where $\{\mathbf{x}^{\boldsymbol{\theta}}_{k-1,i}, W^{\boldsymbol{\theta}}_{k-1,i}\}_{i=1}^N$ are N particles with their associated weights. The joint probability of state and auxiliary variable in (7) can be expressed as:

$$P(\mathbf{x}_k, \zeta_k | \mathbf{y}_{1:k}, \boldsymbol{\theta}) \propto W^{\boldsymbol{\theta}}_{k-1,\zeta_k} P(\mathbf{x}_k | \mathbf{x}_{k-1,\zeta_k}, \boldsymbol{\theta})\, P(\mathbf{y}_k | \mu_{k,\zeta_k}, \boldsymbol{\theta}), \tag{9}$$

for $\zeta_k = 1, \ldots, N$. Note that $\mu_{k,i}$ denotes the state at time step k, i.e., $\mathbf{x}_k$, given the particles $\mathbf{x}^{\boldsymbol{\theta}}_{k-1,i}$. This paper employs the most probable state as this representation, where the $\mu_{k,\zeta_k=i}$ can be expressed as:

$$\mu^{\boldsymbol{\theta}}_{k,i} = \underset{l \in \{1,\ldots,2^n\}}{\operatorname{argmax}} P(\mathbf{x}_k = \mathbf{x}^l \mid \mathbf{x}^{\boldsymbol{\theta}}_{k-1,i}, \boldsymbol{\theta}), \tag{10}$$

for $i = 1, \ldots, N$. The particles and weights representing the state distribution at time step k can be obtained using a two-step update. By employing (10), the weights for the joint state and auxiliary, known as the weights of the *first-stage*, are determined as follows:

$$V^{\boldsymbol{\theta}}_{k,i} = P(\mathbf{y}_k \mid \mu_{k,i}, \boldsymbol{\theta}) W^{\boldsymbol{\theta}}_{k-1,i}, \tag{11}$$

for $i = 1, \ldots, N$. In the second stage, the auxiliary variables $\{\zeta^{\theta}_{k,i}\}_{i=1}^N$, which represent the selected particle indices, are acquired by drawing a sample from the discrete distribution established by $\{V^{\boldsymbol{\theta}}_{k,i}\}_{i=1}^N$ subsequent to appropriate normalization. That is, $\{\zeta^{\theta}_{k,i}\}_{i=1}^N \sim \text{Cat}(\{V^{\boldsymbol{\theta}}_{k,i}\}_{i=1}^N)$. "Cat" here represents the discrete categorical distribution.

To remove the auxiliary variable and create the next particles, one needs to generate N particles at time step k according to the particles and auxiliary variables as:

$$\mathbf{x}^{\boldsymbol{\theta}}_{k,i} \sim P(\mathbf{x}_k \mid \mathbf{x}^{\boldsymbol{\theta}}_{k-1,\zeta_{k,i}}, \boldsymbol{\theta}), \text{ for } i = 1, \ldots, N. \tag{12}$$

The *second stage* weights then can be expressed as:

$$\tilde{W}^{\boldsymbol{\theta}}_{k,i} = \frac{P(\mathbf{y}_k \mid \mathbf{x}^{\boldsymbol{\theta}}_{k,i}, \boldsymbol{\theta})}{P(\mathbf{y}_k \mid \mu^{\boldsymbol{\theta}}_{k,\zeta_{k,i}}, \boldsymbol{\theta})}. \tag{13}$$

The next particles and their associated weights are shown using $\{\mathbf{x}^{\boldsymbol{\theta}}_{k,i}, W^{\boldsymbol{\theta}}_{k,i}\}_{i=1}^N$, where $W^{\boldsymbol{\theta}}_{k,i} = \tilde{W}^{\boldsymbol{\theta}}_{k,i} / \sum_{j=1}^N \tilde{W}^{\boldsymbol{\theta}}_{k,j}$, for $i = 1, \ldots, N$. The unbiased estimator of the likelihood elements at each time step can be obtained as [Pitt (2002)]:

$$P(\mathbf{y}_k \mid \mathbf{y}_{1:k-1}, \boldsymbol{\theta}) \approx \left(\frac{1}{N} \sum_{i=1}^{N} V^{\boldsymbol{\theta}}_{k,i}\right) \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{W}^{\boldsymbol{\theta}}_{k,i}\right). \tag{14}$$

Using (4) and (14), the log-likelihood gets approximated according to the two-stage weights as:

$$L_T(\boldsymbol{\theta}) \approx \sum_{k=1}^{T} \log\left(\frac{1}{N} \sum_{i=1}^{N} V^{\boldsymbol{\theta}}_{k,i}\right) \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{W}^{\boldsymbol{\theta}}_{k,i}\right). \tag{15}$$

The complexity of the APF method for each log-likelihood evaluation is $O(N \times T)$. A large particle number (i.e. N) provides a more accurate approximation of the log-likelihood function and smaller N offers approximations with higher variablities.

Using the computed weights of the particle filter, for any $\boldsymbol{\theta} \in \Theta$ one can write: $\mathbb{E}[\mathbf{x}_k \mid \mathbf{y}_{1:k}, \boldsymbol{\theta}] \approx \sum_{i=1}^N W^{\boldsymbol{\theta}}_{k,i} \mathbf{x}^{\boldsymbol{\theta}}_{k,i}$. The ML estimate of the state given the observations for model $\hat{\boldsymbol{\theta}}^{\text{ML}}$ can be approximated according to (6) as:

$$\hat{\mathbf{x}}^{\text{ML}}_k = \overline{\mathbb{E}[\mathbf{x}_k \mid \mathbf{y}_{1:k}, \hat{\boldsymbol{\theta}}^{\text{ML}}]} \approx \overline{\sum_{i=1}^{N} W^{\hat{\boldsymbol{\theta}}^{\text{ML}}}_{k,i} \mathbf{x}^{\hat{\boldsymbol{\theta}}^{\text{ML}}}_{k,i}}. \tag{16}$$

$\hat{\mathbf{x}}^{\text{ML}}_k$ in (16), ensures approximate MMSE optimality relative to the ML estimate of the model, $\hat{\boldsymbol{\theta}}^{\text{ML}}$.

3.2 Bayesian Optimization for ML Estimation

Assume that Θ forms a space including continuous, discreet, or a mix of continuous and discrete parameters. The goal is to locate the maximum value of the log-likelihood function in (3) with minimal approximate log-likelihood evaluations in (15). To achieve this, Gaussian process (GP) regression [Rasmussen and et al. (2006)] is adopted to capture the log-likelihood function efficiently. The GP model offers a Bayesian and sample-efficient representation of the approximated log-likelihood function.

While the log-likelihood function is deterministic given a set of observed measurements, the particle-based evaluation of the log-likelihood function is stochastic, coming from the variability in the distribution of the particles. The smaller number of particles leads to more stochasticity, whereas larger particle sizes provide less stochastic outcomes (i.e., values closer to the exact log-likelihood function). This paper accounts for this stochasticity and models the log-likelihood function using the following GP regression:

$$L_T(\boldsymbol{\theta}) \approx \mathcal{L}_T(\boldsymbol{\theta}) + \Delta L^N, \tag{17}$$

where $\Delta L^N \sim \mathcal{N}(0, \sigma^2_N)$ is the zero-mean Gaussian residual with variance σ^2_N, representing the stochasticity due to the particle filtering approximation with N particles, and $\mathcal{L}_T(\boldsymbol{\theta})$ is the GP model as:

$$\mathcal{L}_T(\boldsymbol{\theta}) = \mathcal{GP}\left(\mu(\boldsymbol{\theta}), k(\boldsymbol{\theta}, \boldsymbol{\theta})\right). \tag{18}$$

Here, $\mu(.)$ denotes the mean function and $k(.,.)$ signifies the kernel function. One possibility for the mean function $\mu(.)$ in (18), which defines the initial shape of the approximated log-likelihood function across the parameter space, is the constant mean function. The kernel function models the correlation among approximated log-likelihood values across the samples from the space of parameters. The definition of the kernel function is as follows:

$$k(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sigma^2_f \exp\left(-\sum_{i=1}^{|\boldsymbol{\theta}|} \frac{(\boldsymbol{\theta}(i) - \boldsymbol{\theta}'(i))^2}{l_i}\right), \tag{19}$$

where $|\mathbf{v}|$ denotes the size of vector $\mathbf{v}$, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are arbitrary parameter vectors, l_i represents the length-scale hyperparameter linked to the ith parameter, and σ_f^2 is the scale factor hyperparameter, reflecting the correlation between the parameter vectors. The GP's hyperparameters ($\mu(.)$, σ_f, l_i and σ_N) are learned by iteratively optimizing the GP model's marginal likelihood function.

Let $\boldsymbol{\theta}_{1:t}$ denote the initial t samples drawn from the parameter space, with corresponding approximated log-likelihood values $l_{1:t} = [l(\boldsymbol{\theta}_1), ..., l(\boldsymbol{\theta}_t)]$. Utilizing this information, we establish the posterior distribution of the GP model as follows:

$$\mathcal{L}_T(\boldsymbol{\theta}) \mid \boldsymbol{\theta}_{1:t}, l_{1:t} \sim \mathcal{N}(\mu_{\boldsymbol{\theta}}^t, \Sigma_{\boldsymbol{\theta}}^t), \tag{20}$$

where $\mu_{\boldsymbol{\theta}}^t$ and $\Sigma_{\boldsymbol{\theta}}^t$ are defined as:

$$\begin{aligned} \mu_{\boldsymbol{\theta}}^t &= \mu(\boldsymbol{\theta}) + \mathbf{K}_{(\boldsymbol{\theta},\boldsymbol{\theta}_{1:t})}\, \mathbf{K}^{-1}_{(\boldsymbol{\theta}_{1:t},\boldsymbol{\theta}_{1:t})}\, (l_{1:t} - \boldsymbol{\mu}(\boldsymbol{\theta}_{1:t})), \\ \Sigma_{\boldsymbol{\theta}}^t &= k_{(\boldsymbol{\theta},\boldsymbol{\theta})} - \mathbf{K}_{(\boldsymbol{\theta},\boldsymbol{\theta}_{1:t})}\, \mathbf{K}^{-1}_{(\boldsymbol{\theta}_{1:t},\boldsymbol{\theta}_{1:t})}\, \mathbf{K}^{T}_{(\boldsymbol{\theta},\boldsymbol{\theta}_{1:t})}, \end{aligned} \tag{21}$$

and

$$\begin{aligned} \boldsymbol{\mu}(\boldsymbol{\theta}_{1:t}) &= [\mu(\boldsymbol{\theta}_1), ..., \mu(\boldsymbol{\theta}_t)]^T, \\ \mathbf{K}_{(\boldsymbol{\Theta},\boldsymbol{\Theta}')} &= \begin{bmatrix} k(\boldsymbol{\theta}_1, \boldsymbol{\theta}'_1) & \dots & k(\boldsymbol{\theta}_1, \boldsymbol{\theta}'_n) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{\theta}_m, \boldsymbol{\theta}'_1) & \dots & k(\boldsymbol{\theta}_m, \boldsymbol{\theta}'_n) \end{bmatrix}, \end{aligned} \tag{22}$$

for $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_m\}$, $\boldsymbol{\Theta}' = \{\boldsymbol{\theta}'_1, ..., \boldsymbol{\theta}'_n\}$.

Using the GP model's posterior distribution, a sample-efficient sequential optimization is achieved through:

$$\begin{aligned} \boldsymbol{\theta}_{t+1} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmax}} \big(\mu_{\boldsymbol{\theta}}^t - L_{\max}^t\big)\, \Phi\!\left(\big(\mu_{\boldsymbol{\theta}}^t - L_{\max}^t\big)/\sqrt{\Sigma_{\boldsymbol{\theta}}^t}\right) \\ + \sqrt{\Sigma_{\boldsymbol{\theta}}^t}\, \phi\!\left(\big(\mu_{\boldsymbol{\theta}}^t - L_{\max}^t\big)/\sqrt{\Sigma_{\boldsymbol{\theta}}^t}\right), \end{aligned} \tag{23}$$

where the maximization is over the acquisition function at iteration t. This acquisition function is called the Expected Improvement (EI) [Jones et al. (1998); Manshour et al. (2023)], and it serves as a common acquisition function in Bayesian optimization. Note that the acquisition function relies on surrogate (i.e., GP) model predictions instead of the approximated log-likelihood function. The sequential selection in (23) ensures a balance between exploration and exploitation in the parameter estimation process. This process continues until a predefined number of likelihood evaluations are completed or consecutive iterations fail to reveal significant changes in the maximum value of the GP model. Upon completion, the parameter vector yielding the highest assessed log-likelihood value represents the ML estimate of the parameters, formulated as:

$$\hat{\boldsymbol{\theta}}^{\mathrm{ML}} := \boldsymbol{\theta}_{m^*}, \text{ where } m^* = \underset{m\in\{1,...,M\}}{\operatorname{argmax}}\ l_m\,, \tag{24}$$

with M denoting the number of log-likelihood evaluations.

4. NUMERICAL EXPERIMENTS

4.1 Mammalian Cell-Cycle Network

We start our analysis by examining our proposed approach on the mammalian cell-cycle network displayed in Fig. 1. This network manages the process of cellular division in mammals, and its behavior significantly influences overall organismal growth. Comprised of 10 genes, the network's state at time step k is represented by $\mathbf{x}_k = [\mathbf{x}_k(1), ..., \mathbf{x}_k(10)]^T$ =[CycD, Rb, p27, E2F, CycE, CycA, Cdc20, Cdh1, UbcH10, CycB]^T. Each gene's state, denoted by $\mathbf{x}_k(i) \in \{0, 1\}$, indicates whether the ith gene

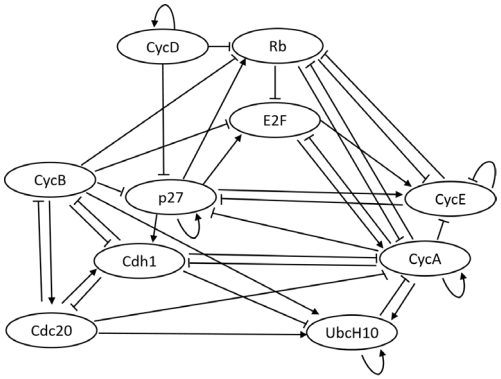

Fig. 1. Mammalian Cell-Cycle network pathway diagram.

is activated (1) or inactive (0). The state process for this network can be expressed as [Hosseini and Imani (2024)]:

$$\mathbf{x}_k \sim \overline{\mathbf{C}^{\boldsymbol{\theta}}\mathbf{x}_{\mathbf{k-1}}} \oplus \mathbf{n}_k\,, \tag{25}$$

where the operator $\overline{\mathbf{v}}$ assigns 0 to vector elements less than or equal to 0.5 and 1 to others, the symbol "$\oplus$" denotes componentwise module-2 addition, and $\mathbf{n}_k$ represents the independent and identically distributed (i.i.d.) binary noise, where the probability of the i^{th} element of $\mathbf{n}_k$ being equal to 1 is p. The parameter p lies between 0 and 0.5, and indicates the stochasticity in the state process where $p = 0$ signifies no stochasticity and $p = 0.5$ represents maximum stochasticity. The connectivity matrix, $\mathbf{C}^{\boldsymbol{\theta}}$ for the mammalian cell-cycle network can be represented using Fig. 1 by the following 10×10 matrix:

$$\mathbf{C}^{\boldsymbol{\theta}^*} = \begin{bmatrix} +1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & +1 & +1 & +1 & +1 & 0 & 0 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & +1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 & -1 & +1 & 0 & -1 & +1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & +1 & +1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & +1 & 0 \\ 0 & -1 & -1 & -1 & 0 & 0 & +1 & -1 & +1 & 0 \end{bmatrix}^T, \tag{26}$$

where the elements +1 and -1 indicate positive and suppressive regulations, and 0 denotes no relationships between the genes. The value at the ith row and jth column ($c_{i,j}$) denotes the type of regulation or interaction occurring from the jth gene to the ith gene. Note that $\boldsymbol{\theta}^*$ in $\mathbf{C}^{\boldsymbol{\theta}^*}$ denotes the true model of the regulatory network.

The genes' states are only partially observable through the gene-expression data. The widely recognized Gaussian model, frequently applied in live-cell imaging-based assays and cDNA microarrays [Hua et al. (2012)], characterizes the measurement process in the POMDP model in (2) as follows:

$$\mathbf{y}_k(i) = m + \delta \mathbf{x}_k(i) + \mathbf{v}_k(i)\,,\ k = 1, 2, \dots, T, \tag{27}$$

where $\mathbf{v}_k(i) \sim \mathcal{N}(0, \sigma^2)$ represents a zero-mean Gaussian noise vector with uncorrelated elements, m is the baseline expression (referring to the "zero" state), and δ denotes the differential expression value.

The following parameters are used for all the experiments: trajectory length $T = 100$, process noise $p = 0.05$, baseline expression $m = 20$, and differential expression $\delta = 30$. Moreover, all the experiments are repeated 10 times, and all the figures display the mean values and the 95% confidence intervals for the results. For comparison purposes, the same experiment is carried out using the Simulated

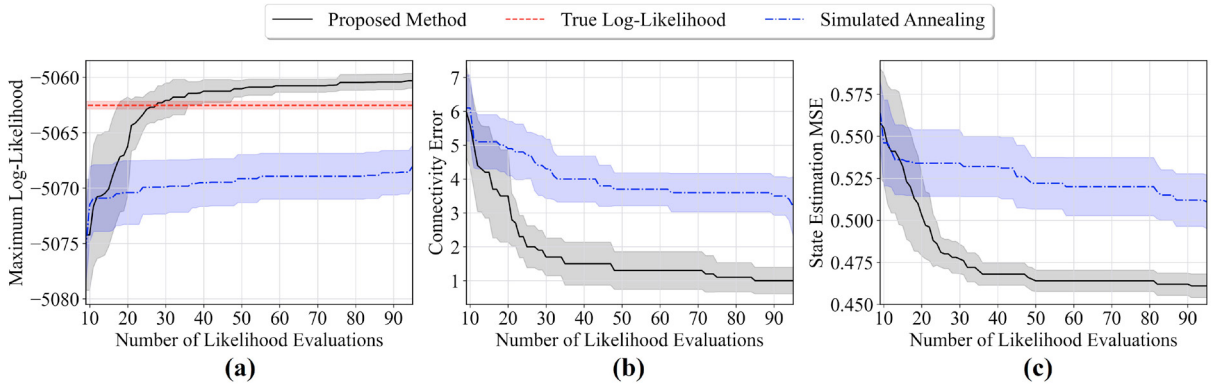

Fig. 2. Performance comparison of our method and simulated annealing throughout topology optimization of the mammalian cell-cycle network: (a) average log-likelihood progress (b) average connectivity error progress (c) average state estimation MSE progress.

Annealing [Biswas and Acharyya (2014)] method, which has been extensively used for efficiently optimizing parameters across various parameter spaces, particularly in the context of biological networks [Handzlik and Manu (2022)].

We consider that the following 10 elements of the true connectivity matrix to be unknown: $c_{2,1} = -1$, $c_{3,5} = -1$, $c_{3,10} = -1$, $c_{4,2} = -1$, $c_{5,4} = +1$, $c_{6,7} = -1$, $c_{6,9} = -1$, $c_{8,3} = +1$, $c_{9,6} = +1$, $c_{9,8} = -1$. Since each of these unknown interactions could take values in $\{-1, 0, +1\}$, there will be $3^{10} = 59,049$ possible models, Θ, for the mammalian cell-cycle network.

The proposed method with particle size $N = 5,000$ is implemented for this problem. Fig. 2(a) shows the average maximum log-likelihood via our method and simulated annealing after 100 log-likelihood evaluations. The solid black curve is our method's log-likelihood, the blue dotted one is from simulated annealing, and the dashed red line is the true log-likelihood, approximated using the particle filter. One can see that our method consistently achieves a higher log-likelihood. Fig. 2(b) displays the average connectivity error during optimization, measuring the L_1 norm of the difference between true and estimated vectorized connectivity matrices. Our method maintains a smaller connectivity error compared to simulated annealing in all the optimization steps. Additionally, Fig. 2(c) examines how our method performs concerning the average state estimation MSE, defined as $\sum_{k=1}^{T} \|\mathbf{x}_k - \hat{\mathbf{x}}_k^{\mathrm{ML}}\|_1/T$. Fig. 2(c), shows that our method yields lower state estimation MSE throughout the inference process, compared to simulated annealing.

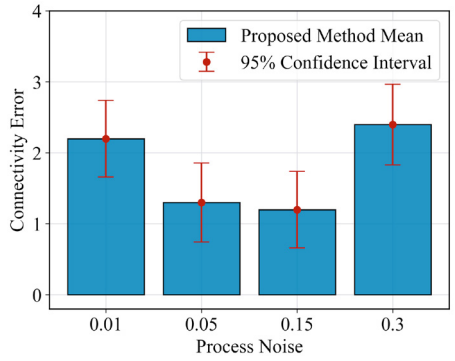

Fig. 3. Changes of connectivity error in the presence of different process noise (p) in the cell-cycle network.

In this part, the impact of varying process noise (p) on connectivity error is studied. In Fig. 3 four p values (0.01, 0.05, 0.15, and 0.3) are examined, showcasing trends in the average connectivity error after 100 likelihood evaluations. Notably, errors decrease at p values of 0.05 and 0.15 but rise again at 0.3. At low p (0.01), the network tends to stabilize in attractor states, complicating the accurate inference of the network model and resulting in a higher connectivity error. Conversely, higher p values (0.05, 0.15) introduce more stochasticity, aiding in exploring diverse system states and improving inference. However, beyond a certain point (0.3), excessive stochasticity negatively affects the inference of gene interactions, leading to increased connectivity error due to the system's unpredictability.

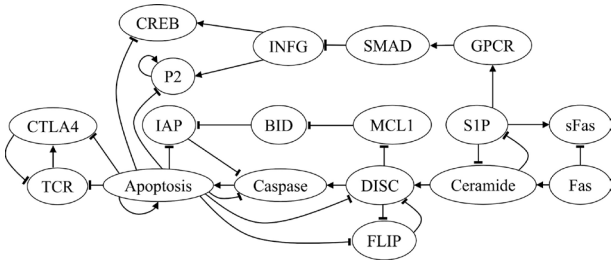

Fig. 4. T-LGL leukemia network pathway diagram

4.2 T-cell Large Granular Lymphocyte Leukemia Network

T-cell large granular lymphocyte (T-LGL) leukemia is a long-lasting illness marked by clonal proliferation of cytotoxic T cells. A simplified Boolean network model of T cell survival signaling in the context of this disease is constructed in Saadatpour et al. (2011). Fig. 4 represents the pathway diagram for this network. T-LGL leukemia's state at time step k can be shown by $\mathbf{x}_k = [\mathbf{x}_k(1), ..., \mathbf{x}_k(18)]^T$ =[CTLA4, TCR, CREB, INFG, P2, GPCR, SMAD, FAS, sFas, Ceramide, DISC, Caspase, FLIP, BID, IAP, MCL1, S1P, Apoptosis]^T. The following 10 interactions of the T-LGL leukemia network are assumed to be unknown: $c_{2,1} = -1$, $c_{3,4} = +1$, $c_{10,8} = +1$, $c_{16,11} = -1$, $c_{11,13} = -1$, $c_{12,14} = +1$, $c_{9,17} = +1$, $c_{2,18} = -1$, $c_{13,18} = -1$, $c_{18,18} = -1$. These 10 unknown elements in the connectivity matrix, lead to 59,049 possible models.

The exact computation of log-likelihood in this network is extremely expensive, however using our method, the complexity significantly reduces. The proposed method is applied to this problem using 300,000 particles for 100 likelihood evaluations and the results are compared with simulated annealing. Fig. 5(a) represents the progress of log-likelihood over the 100 evaluations. As can be seen in the figure, the average maximum log-likelihood using the proposed method is more than the values obtained using simulated annealing. Moreover, Fig. 5(b) and 5(c), show

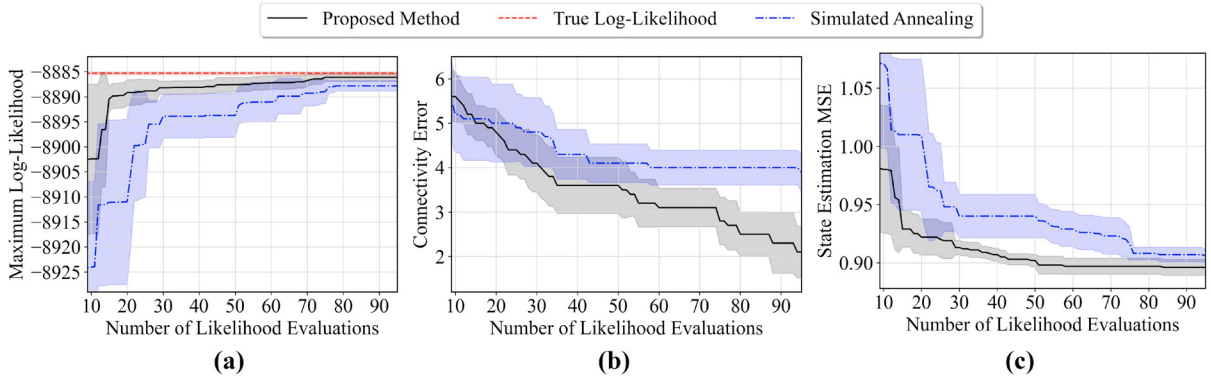

Fig. 5. Performance comparison of our method and simulated annealing throughout topology optimization of the T-LGL leukemia network: (a) average log-likelihood progress (b) average connectivity error progress (c) average state estimation MSE progress.

the progress of connectivity error and state estimation MSE for this experiment. One can see that in both figures, the proposed method has lower values than simulated annealing. Therefore, these figures demonstrate the superior performance of our approach during the inference process.

5. CONCLUDING REMARKS

This paper presented a novel approach to tackle the computational challenges in inferring partially observed Boolean dynamical systems (POBDS), which are crucial for modeling diverse systems like genomics systems and network security. By introducing a kernel-based particle filtering method, the paper mitigates the scalability issues inherent in these models. The Gaussian process (GP) is leveraged to approximate the computationally expensive likelihood function and Bayesian optimization is utilized for sample-efficient explorations over the parameter spaces. The application of this method was demonstrated on two biological networks showcasing its potential in addressing complex real-world problems. The obtained results showed that the proposed method enables scalable and efficient inference for POBDS. Our future work will include the inference in domains with extremely large parameter spaces, which demands more sequential search and scalable alternatives to the GP model.

REFERENCES

Alali, M. and Imani, M. (2022). Inference of regulatory networks through temporally sparse data. *Frontiers in Control Engineering*, 3.

Alali, M. and Imani, M. (2024). Bayesian lookahead perturbation policy for inference of regulatory networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1–14.

Biswas, S. and Acharyya, S. (2014). Gene expression profiling by estimating parameters of gene regulatory network using simulated annealing: A comparative study. In *2014 IEEE International Advance Computing Conference (IACC)*, 56–61.

Davidich, M.I. and Bornholdt, S. (2008). Boolean network model predicts cell cycle sequence of fission yeast. *PloS one*, 3(2), e1672.

Handzlik, J.E. and Manu (2022). Data-driven modeling predicts gene regulatory network dynamics during the differentiation of multipotential hematopoietic progenitors. *PLOS Computational Biology*, 18(1), 1–31.

Hosseini, S.H. and Imani, M. (2024). An optimal Bayesian intervention policy in response to unknown dynamic cell stimuli. *Information Sciences*, 666, 120440.

Hua, J., Sima, C., Cypert, M., Gooden, G.C., Shack, S., Alla, L., Smith, E.A., Trent, J.M., Dougherty, E.R., and Bittner, M.L. (2012). Dynamical analysis of drug efficacy and mechanism of action using GFP reporters. *Journal of Biological Systems*, 20(04), 403–422.

Imani, M. and Braga-Neto, U. (2018). Particle filters for partially-observed Boolean dynamical systems. *Automatica*, 87, 238–250.

Imani, M. and Braga-Neto, U.M. (2017). Maximum-likelihood adaptive filter for partially observed Boolean dynamical systems. *IEEE Transactions on Signal Processing*, 65(2), 359–371.

Imani, M., Dougherty, E.R., and Braga-Neto, U.M. (2019). Boolean Kalman filter and smoother under model uncertainty. *Automatica*.

Jones, D.R., Schonlau, M., and Welch, W.J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4), 455–492.

Kazeminajafabadi, A. and Imani, M. (2023). Optimal monitoring and attack detection of networks modeled by bayesian attack graphs. *Cybersecurity*, 6(1), 22.

Manshour, N., He, F., Wang, D., and Xu, D. (2023). Integrating protein structure prediction and bayesian optimization for peptide design. In *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*.

Pitt, M.K. (2002). Smooth particle filters for likelihood evaluation and maximisation. Technical report, University of Warwick, Department of Economics.

Pitt, M.K. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American statistical association*, 94(446), 590–599.

Puvsnik, vZ., Mraz, M., Zimic, N., and Movskon, M. (2022). Review and assessment of Boolean approaches for inference of gene regulatory networks. *Heliyon*.

Rasmussen, C.E. and et al. (2006). *Gaussian processes for machine learning*. MIT Press.

Ravari, A., Ghoreishi, S.F., and Imani, M. (2024). Optimal inference of hidden Markov models through expert-acquired data. *IEEE Transactions on Artificial Intelligence*, 1–15.

Saadatpour, A., Wang, R.S., Liao, A., Liu, X., Loughran, T.P., Albert, I., and Albert, R. (2011). Dynamical and structural analysis of a t cell survival network identifies novel candidate therapeutic targets for large granular lymphocyte leukemia. *PLoS computational biology*, 7(11), e1002267.

Zhang, K. (2023). A survey on observability of Boolean control networks. *Control Theory and Technology*, 1–33.