Graph-Based Bayesian Optimization for Large-Scale Objective-Based Experimental Design

Mahdi Imani, Seyede Fatemeh Ghoreishi

***Abstract*—Design is an inseparable part of most scientific and engineering tasks, including real and simulation-based experimental design processes and parameter/hyperparameter tuning/optimization. Several model-based experimental design techniques have been developed for design in domains with partial available knowledge about the underlying process. This paper focuses on a powerful class of model-based experimental design called *the mean objective cost of uncertainty* (MOCU). The MOCU-based techniques are objective-based, meaning that they take the main objective of the process into account during the experimental design process. However, the lack of scalability of MOCU-based techniques prevents their application to most practical problems, including large discrete or combinatorial spaces. To achieve a scalable objective-based experimental design, this paper proposes a graph-based MOCU-based Bayesian optimization framework. The correlations among samples in the large design space are accounted using a graph-based Gaussian process, and an efficient closed-form sequential selection is achieved through the well-known expected improvement policy. The proposed framework's performance is assessed through the structural intervention in gene regulatory networks, aiming to make the network away from the states associated with cancer.**

***Index Terms*—Model-Based Experimental Design, Mean Objective Cost of Uncertainty (MOCU), Graphical Models, Bayesian Optimization, Gene Regulatory Networks.**

I. INTRODUCTION

Design is a key element of most scientific and engineering tasks, including *real experimental design settings,* such as materials/drugs design [1], and aerospace engineering [2], [3]; *simulation-based experimental design,* such as design in robotics [4], and transportation [5]; and *parameter learning or hyper-parameter tuning,* such as efficient tuning of the parameters of Markov chain Monte Carlo (MCMC) methods [6], Bayesian networks [7]–[9], and deep neural networks [10]. The advancements in technology have highly expanded the size of systems and phenomena that are under study.

Experimental design can be categorized into two broad classes of model-based and data-driven techniques. Model-based experimental design techniques have significantly impacted several science and engineering applications [11]–[17]. These techniques aim to select proper designs according to the partial knowledge about the underlying systems/processes, and have been successfully applied in various domains including parameter inference, prediction, or model discrimination [18], as well as real experimental processes such as materials science [19], genomics and metagenomics [20]–[22], and aerospace engineering [3], [23], [24]. Information theoretic-based [25], [26] and objective-based [12], [27], [28] techniques are two popular classes of model-based experimental design techniques, where the former techniques aim to reduce the uncertainty in the whole system/process, whereas the latter ones aim to reduce the uncertainty that matters the most with respect to the main objective of the process.

The main focus of this paper is the mean objective cost of uncertainty (MOCU) [12], [27], [28], which is the most popular policy in the class of objective-based experimental design techniques. Despite the success of the existing MOCU-based experimental design techniques, the main challenges rendering intractable their application to most practical domains are:

- *Large Design Spaces:* many real experimental design settings are in large discrete/combinatorial design spaces. This makes searching over all possible design inputs and selecting the best one an impossible or computationally-expensive task.
- *Computations Associated with the Selection Process:* evaluating the effectiveness of each design input is a big challenge in most practical problems. The complexity of evaluation increases exponentially with the size of systems and the amount of uncertainty, which limits or makes impossible the selection process in most realistic scenarios.
- *Huge Cost of Real Experiments:* the huge cost associated with taking real experiments limits the total number of experiments to take in many practical domains. The existing model-based experimental design techniques often come short during the efficient selection of costly experiments in systems with large design inputs and huge uncertainty.

These challenges mentioned above abound in domains such as cyber-physical systems for finding the type of interactions between components through huge uncertainty sources for making reliable decisions, and smart cities for selecting the best locations for bike or bus stations, parking, shopping centers, new highways, or roads, bridges, traffic signs, and surveillance cameras for enhancing social welfare, boosting the traffic flow, and increasing the safety of our communities. To expand the MOCU-based experimental design techniques to domains with large discrete or combinatorial spaces, this paper develops a graph-based Bayesian optimization MOCU-based experimental design (GBO-MOCU) framework. The proposed framework is capable of selecting the minimum number of experiments that reveal the highest information about the underlying process. The proposed framework constructs a graph-based Gaussian process surrogate model representing

M. Imani is with the Department of Electrical and Computer Engineering at the George Washington University, and S. F. Ghoreishi is with the Institute for Systems Research (ISR) at the University of Maryland, `mimani@gwu.edu, sfg@umd.edu`

the MOCU-related cost function over the design space. An efficient sequential search over the design space is achieved through the well-known expected improvement policy [29], [30]. The proposed framework offers several benefits, including:

- *Scalable design* in large discrete or combinatorial spaces and systems/processes with a huge amount of uncertainty, where exhaustive exploration of all options is intractable or inefficient;
- *Fast selection* of experiments in large domains with tight time-limit constraints by enabling the incorporation of arbitrary Bayesian optimization techniques for the selection process;
- *Efficient experimental design* in domains with expensive/costly experiments, in which the total number of experiments to take is highly limited.

The aforementioned features (i.e., scalability, fast decision making, and efficiency) distinct the proposed framework from a broader range of model-based experimental design techniques.

A primary goal in genomics studies is to find proper treatments for chronic diseases such as cancer. Genomics signal processing often models genomics systems as networks, known as gene regulatory networks, and tries to derive intervention strategies to shift these systems' undesirable behavior to desirable ones. This can be seen as finding ways to shift the dynamics of gene regulatory networks away from carcinogenic states [31]–[33]. Structural intervention [34] comprises a wide class of intervention techniques, which aim to make a single change in the network structure (e.g., alter the interaction between genes) in such a way that the maximum reduction in the probability of observing undesirable (e.g., cancerous) states occur. The optimal solution for this type of intervention, which is based on the Markov properties of the dynamics of gene regulatory networks, is provided in [20].

Despite huge progress in deriving efficient intervention strategies for gene regulatory networks, the large amount of uncertainty in these systems often leads to the poor performance of the intervention. The uncertainty, which arises from the complexity of genomics systems, appears in terms of unknown interactions between genes in networks' topology. This necessitates the development of proper experimental design strategies to reduce the amount of uncertainty in these systems so that the highest performance of the intervention process is achieved. Toward this, this project will employ the proposed graph-based MOCU-based experimental design policy for optimal sequential experimental design in the context of structural intervention in large gene regulatory networks with a large amount of uncertainties.

The structural intervention in regulatory networks deals with making optimal changes in the structure of fully-known GRNs. However, there are often huge sources of uncertainty in regulatory networks; for instance, certain regulatory relations are unknown, or at least not known with certainty. In these scenarios, it is highly critical to reduce the uncertainty in regulatory networks to achieve better performance in the intervention process. This paper demonstrates the application of the proposed graph-based MOCU-based experimental design policy in the context of structural intervention of large GRNs with a large amount of uncertainties.

The article is organized as follows. In section II, the model-based experimental design is described. The proposed graph-based Bayesian optimization MOCU-based experimental design is formulated in Section III. Experimental design for structural intervention in genomics is described in Section IV. Section V presents results for the numerical experiments with random Boolean networks and the mammalian cell-cycle gene regulatory network. Finally, section VI contains the concluding remarks.

II. MODEL-BASED EXPERIMENTAL DESIGN — MEAN OBJECTIVE COST OF UNCERTAINTY (MOCU)

Model-based experimental design has a very long history in science and engineering [11], [35]. Two well-known classes of them are information theoretic-based [25], [26], [36], [37] and objective-based techniques [12], [28]. These techniques require the existence of rich prior knowledge about the underlying process. However, in most practical problems, the experiments are often so costly, either economically or computationally, necessitating taking the minimum number of costly experiments to achieve the desired performance.

Entropy-based experimental design refers to a wide class of techniques that aim to take experiments that maximally reduce the uncertainty in systems/processes [13]. By contrast, the mean objective cost of uncertainty (MOCU) [12], [27], [28] is the most well-known objective-based experimental design policy that quantifies the uncertainty based on the degree to which the model uncertainty affects the objective. The simplest MOCU policy, which aims to sequentially take experiments to maximally reduce the MOCU values, is described in the following paragraphs.

Suppose that the uncertainty in the system/process is encoded in a parameter vector $\boldsymbol{\theta}$, with $P(\boldsymbol{\theta})$ indicates the prior distribution over the parameter space Θ. Let $\mathbb{A}$ denote the action space and $\xi_{\boldsymbol{\theta}}(\mathbf{a})$ be the cost of model parameterized by $\boldsymbol{\theta}$ and action $\mathbf{a} \in \mathbb{A}$. The actions and action-outcomes (i.e., costs associated with actions) differ from experiments and experiment-outcomes. In fact, action and action-outcome depend on the main objective of the process: for instance, in classification, estimation, or filtering, the action could be selecting features, sensors, or biomarkers, and action-outcome could be classification, estimation, or filtering error. For a fully-known system, the best action should lead to the smallest cost value (i.e., action-outcome). However, the system is often uncertain, and the cost is defined over the uncertainty class.

Let $\mathcal{D}$ be the design/experiment space. The way that each experiment $\mathbf{d} \in \mathcal{D}$ impacts the main objective is through reducing the amount of uncertainty in the systems, in order to maximally improve the performance of the main objective of the process. Let $\mathbf{d}_{1:t} = (\mathbf{d}_1, ..., \mathbf{d}_t)$ and $y_{1:t} = (y_1, ..., y_t)$ be the sequences of taken experiments and their corresponding outcomes up to step t. The posterior distribution can be computed as:

$$P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}) = \frac{p_{\boldsymbol{\theta}}(\mathbf{d}_{1:t}, y_{1:t})\, P(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}' \in \Theta} p_{\boldsymbol{\theta}'}(\mathbf{d}_{1:t}, y_{1:t})\, P(\boldsymbol{\theta}')\, d\boldsymbol{\theta}'}\,, \tag{1}$$

for $\boldsymbol{\theta} \in \Theta$, where the integral in denominator of (1) should be replaced by summation in cases with discrete parameter spaces. The way that experiments impact the posterior depends on the applications; for instance experiments could reduce the uncertainty or create some new sets of data.

Letting $P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t})$ to be the posterior probability of model $\boldsymbol{\theta}$ given $(\mathbf{d}_{1:t}, y_{1:t})$, the *optimal Bayesian* (OB) action can be defined as [38], [39]:

$$\mathbf{a}_{\mathrm{OB}} = \underset{\mathbf{a} \in \mathbb{A}}{\operatorname{argmin}} \, \mathbb{E}_{\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}} [\xi_{\boldsymbol{\theta}}(\mathbf{a})], \tag{2}$$

where the expectation is over the uncertainty represented in the posterior distribution of parameter $\boldsymbol{\theta}$, and the OB action in (2) yields the minimum expected cost relative to this posterior.

The mean objective cost of uncertainty (MOCU) selects an experiment at each step to reduce the uncertainty that maximally impacts the objective of the process. Let $\mathbf{a}_{\boldsymbol{\theta}}$ be the optimal action for a model defined by uncertainty $\boldsymbol{\theta}$. Then $\xi_{\boldsymbol{\theta}}(\mathbf{a}) \geq \xi_{\boldsymbol{\theta}}(\mathbf{a}_{\boldsymbol{\theta}})$, for any $\boldsymbol{\theta} \in \Theta$ and $\mathbf{a} \in \mathbb{A}$. The MOCU value associated with the posterior distribution of the unknown parameter $P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t})$ and the class of actions $\mathbb{A}$ can be defined by:

$$M_{\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}} = \mathbb{E}_{\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}} \left[\xi_{\boldsymbol{\theta}}(\mathbf{a}_{\mathrm{OB}}) - \xi_{\boldsymbol{\theta}}(\mathbf{a}_{\boldsymbol{\theta}}) \right], \tag{3}$$

where $\mathbf{a}_{\mathrm{OB}}$ is the OB action (defined in (2)) and its value depends on the posterior of the parameter, whereas the optimal action $\mathbf{a}_{\boldsymbol{\theta}}$ is designed for a specific model $\boldsymbol{\theta} \in \Theta$. The MOCU in (3) is the expected increase in cost values, that results from applying an OB action to the entire models represented in Θ instead of applying the optimal action to the unknown true model. The MOCU-based policy [12], [27], [28] selects an experiment $\mathbf{d}$ at each step from the set of experiments $\mathcal{D}$ that results in the maximal reduction in the MOCU value in the next step. This can be written as:

$$\mathbf{d}_{t+1} = \underset{\mathbf{d} \in \mathcal{D}}{\operatorname{argmax}} \, h_t(\mathbf{d}), \tag{4}$$

where $h_t(\mathbf{d})$ is the *MOCU-Reduction* function defined as:

$$h_t(\mathbf{d}) = \mathbb{E}_t \big[M_{\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}} - M_{\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t}, \mathbf{d}_{t+1} = \mathbf{d}} \big], \tag{5}$$

and the expectation in (4) is taken with respect to the posterior probability $P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t})$.

The prior probability over the parameter space depends on the application. In cases that no prior knowledge is available, the non-informative (uniform) prior distribution should be used over the parameter space.

III. PROPOSED FRAMEWORK

A. Difficulties of Design in Large Discrete or Combinatorial Spaces

Let $(\mathbf{d}_{1:t}, y_{1:t})$ be all available information at step t of an experimental design process, where $\mathbf{d}_{1:t}$ and $y_{1:t}$ denote the taken experiments and their outcomes respectively. The goal of MOCU policy is to select the experiment at step $t+1$, i.e., $\mathbf{d}_{t+1}$. This can be done through finding the solution for the maximization problem in (4), which requires computation of the MOCU-Reduction function in (5) for all samples in the design space $\mathcal{D}$. However, this can be computationally very expensive or impossible for most large-scale uncertain practical problems due to the following reasons:

- *Large Systems with Huge Uncertainty:* The computational expense associated with the evaluation of the MOCU-Reduction function in (5) at any single design sample in $\mathcal{D}$ increases exponentially with the size of systems, and amount of uncertainty. This makes the selection by the MOCU policy impossible or computationally very expensive in most practical domains.
- *Large Discrete or Combinatorial Design Spaces:* The design space, i.e., $\mathcal{D}$, in most practical problems is often large discrete or combinatorial, which necessities conducting a large number of experiments to achieve a proper performance in the design process. However, the number of experiments is often limited due to the limitation in resources or the huge cost associated with the experiments, rendering intractable the application of the existing MOCU-based techniques or resulting in their poor performance.

An example of domains in which both of the aforementioned issues are often encountered is genomics/metagenomics [22], where the experiments should be taken in a sequentially and timely manner. For instance, perturbations/excitations in genomics studies should be performed at tight time slots, aligned with the dynamical evolution of these complex systems/processes [40]–[43]. Toward addressing the above issues, in the following paragraphs, the proposed graph-based Bayesian optimization mean-objective cost of uncertainty (GBO-MOCU) framework for scalable and efficient experimental design is described.

B. Graph-Based Gaussian Process for Modeling the MOCU-Reduction Function

Let $h_t(\mathbf{d})$ be the MOCU-Reduction function, which can be approximated by performing a computationally expensive process (see equation (5)) at any given sample $\mathbf{d}$. This MOCU-Reduction function can be expressed as:

$$h_t(\mathbf{d}) \approx \mathcal{H}_t(\mathbf{d}) + \Delta h_t, \tag{6}$$

where $\mathcal{H}_t(\mathbf{d})$ is a Gaussian process (GP) surrogate model representing the MOCU-Reduction function, and Δh_t is a zero-mean Gaussian residual with standard deviation σ_t, which models the uncertainty coming from the approximation made in evaluation of the MOCU-Reduction function.

The prior distribution for the GP model can be expressed as:

$$\mathcal{H}_t(\mathbf{d}) = \mathcal{GP}\left(\mu_t(\mathbf{d}), k_t(\mathbf{d}, \mathbf{d})\right),$$

where $\mu_t(.)$ and $k_t(.,.)$ are the mean and a real-valued kernel function. Since the design takes in a large discrete or combinatorial space, the graph-based kernel function proposed in [44] is considered for modeling the correlation in the MOCU-Reduction function over the design space. We represent the search space as a graph $\mathcal{G} = (V, E)$, where each vertex in V specifies a design input, and an edge in E determines whether the MOCU-Reduction values in two design inputs are considered to be similar or not. Letting $A_{\mathcal{G}}$ be the adjacency matrix and $D_{\mathcal{G}}$ be the degree matrix of a graph (see [45], for more information), the correlation can be assessed through the graph Laplacian as [45]:

$$L(\mathcal{G}) = D_{\mathcal{G}} - A_{\mathcal{G}}. \tag{7}$$

The eigenvalues and eigenvectors of the graph Laplacian can be seen as the Fourier frequencies and bases. Letting $\{\lambda_1, \lambda_2, \cdots, \lambda_{|V|}\}$ and $\{\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_{|V|}\}$ be the eigenvalues and eigenvectors respectively, we can define the diffusion kernel function as:

$$k_t(p,q) = \sum_{i=1}^{n} \exp(\beta_t \lambda_i)\mathbf{u}_i(p)\,\mathbf{u}_i(q), \tag{8}$$

where p and q are two arbitrary nodes of the graph, and β_t is the hyperparameter of the kernel function. Large eigenvalues (i.e., $\lambda_i >> 1$) correspond to high frequencies and model a large dependency between two nodes of the graph. Defining $\Lambda = \text{diag}(\lambda_1, \cdots, \lambda_{|V|})$ and $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_{|V|}]$, the kernel over all nodes of the graph (i.e., the entire design inputs) can be computed as:

$$\mathbf{K} = \mathbf{U} \exp\left(\beta_t \Lambda\right) \mathbf{U}^T. \tag{9}$$

It should be noted that for any two sets of $\mathbf{D} \subset \mathcal{D}$ and $\mathbf{D}' \subset \mathcal{D}$, the kernel function can be denoted by $\mathbf{K}_{\mathbf{D},\mathbf{D}'}$, and can be reconstructed from the kernel function in (9). Notice that the graph Cartesian product can be employed for computation of the kernel in combinatorial spaces [45].

Let $\mathbf{D}_m = (\mathbf{d}^{(1)}, \ldots, \mathbf{d}^{(m)})$ be the set of m design samples from the design space, with the approximated MOCU-Reduction values $\mathbf{h}_m = [h_t(\mathbf{d}^{(1)}), \ldots, h_t(\mathbf{d}^{(m)})]^T$. The posterior distribution of $\mathcal{H}_t(\mathbf{d})$ can be obtained as [46]:

$$\mathcal{H}_t(\mathbf{d}) \mid \mathbf{D}_m, \mathbf{h}_m \sim \mathcal{N}\left(\bar{\mathcal{H}}_t(\mathbf{d}), \text{cov}_t\left(\mathbf{d}, \mathbf{d}\right)\right), \tag{10}$$

where

$$\begin{aligned}
\bar{\mathcal{H}}_t(\mathbf{d}) &= \mu_t(\mathbf{d}) + \mathbf{K}_{\mathbf{d},\mathbf{D}_m}\left(\mathbf{K}_{\mathbf{D}_m,\mathbf{D}_m} + \mathbf{\Sigma}_m\right)^{-1}\left(\mathbf{h}_m - \mu_t(\mathbf{D}_m)\right), \\
\text{cov}_t(\mathbf{d}, \mathbf{d}) &= k_t(\mathbf{d}, \mathbf{d}) - \mathbf{K}_{\mathbf{d},\mathbf{D}_m}\left(\mathbf{K}_{\mathbf{D}_m,\mathbf{D}_m} + \mathbf{\Sigma}_m\right)^{-1}\mathbf{K}^T_{\mathbf{d},\mathbf{D}_m},
\end{aligned} \tag{11}$$

and $\mathbf{\Sigma}_m$ is a diagonal matrix of size m with the ith diagonal element $(\mathbf{\Sigma}_m)_{ii} = \sigma_t^2$, and

$$\mathbf{K}_{\mathbf{D},\mathbf{D}'} = \begin{bmatrix} k_t(\mathbf{d}_1, \mathbf{d}'_1) & \ldots & k_t(\mathbf{d}_1, \mathbf{d}'_r) \\ \vdots & \ddots & \vdots \\ k_t(\mathbf{d}_l, \mathbf{d}'_1) & \ldots & k_t(\mathbf{d}_l, \mathbf{d}'_r) \end{bmatrix}, \tag{12}$$

for $\mathbf{D} = \{\mathbf{d}_1, \ldots, \mathbf{d}_l\}$ and $\mathbf{D}' = \{\mathbf{d}'_1, \ldots, \mathbf{d}'_r\}$. The MOCU-Reduction function before any evaluation is large, represented by a Gaussian process with covariance $k_t(\mathbf{d}, \mathbf{d})$, while at iteration m, this function can be predicted based on the sequence of queried samples $\mathbf{D}_m$ and the approximate MOCU-Reduction values $\mathbf{h}_m$. The uncertainty in the MOCU-Reduction function decreases as more samples from the design space are evaluated and added to the GP.

C. *Hyperparameter Learning*

The hyperparameters of the GP, such as the parameters of the mean and kernel functions, can be estimated at each step. Specifically, for determining β_t, which is crucial for obtaining a flexible kernel function, one can employ the following sampling procedure:

1) Set $r = 0$ and choose a starting $\beta_t^{(r)}$ for which the probability is non-zero.
2) Sample q uniformly from $\left[0, p\left(\beta_t^{(r)} \mid \mathbf{h}_m, \mathbf{D}_m\right)\right]$
3) Draw a new value $\beta_t^{(r+1)}$ uniformly from regions for which $p\left(\beta_t^{(r)} \mid \mathbf{h}_m, \mathbf{D}_m\right) > q$.
4) Repeat from 2 using $\beta_t^{(r+1)}$.

The value of the noise statistics, σ_t^2, which represents the variance in the evaluation of the MOCU-Reduction function can be estimated in the following three possible ways: 1) approximating the MOCU-Reduction function at a single design input several times and measuring the variance of the approximated values; 2) using the available error estimation theoretical upper bounds for the variance of the approximation (e.g., Monte Carlo approximation [47]) in computation of the expectation in (5); 3) learning the noise parameter on the fly using the following marginal likelihood:

$$\mathbf{h}_m \mid \mathbf{D}_m \sim \mathcal{N}\left(\mu_t(\mathbf{D}_m), \mathbf{K}_{\mathbf{D}_m,\mathbf{D}_m}\right) + \mathbf{\Sigma}_m\,. \tag{13}$$

Finally, adaptive constant mean is one possible choice for the mean function $\mu_t(\mathbf{d})$, which can be expressed as $\mu_t(\mathbf{d}) = \min_{i=1,\ldots,m} \mathbf{h}_m(i)$, for $\mathbf{d} \in \mathcal{D}$. This mean function does not have any parameter, and prevents over/under estimating the MOCU-Reduction function.

D. *Bayesian Optimization for Sequential Experimental Design*

Intelligent selection of samples from the large design space is critical in achieving a fast and accurate solution for optimization problem in (4). Let $\mathbf{D}_m = (\mathbf{d}^{(1)}, \ldots, \mathbf{d}^{(m)})$ be m samples from the design space $\mathcal{D}$, with the evaluated MOCU-Reduction functions, $\mathbf{h}_m = [h_t(\mathbf{d}^{(1)}), \ldots, h_t(\mathbf{d}^{(m)})]^T$. Let $h_m^{\max} = \max_{i=1,\ldots,m} \mathbf{h}_m(i)$ be the maximum value of the evaluated MOCU-Reduction function up to iteration m. The *Expected Improvement* (EI) policy evaluates h_t at the sample that, in expectation, increases upon $h_m^{\max}$ the most. This can be formulated as:

$$\mathbf{d}^{(m+1)} = \underset{\mathbf{d} \in \mathcal{D}}{\text{argmax}}\, \alpha^t_{\text{EI}}(\mathbf{d}), \tag{14}$$

where

$$\begin{aligned}
\alpha^t_{\text{EI}}(\mathbf{d}) &= \mathbb{E}_m\left[\max_{\mathbf{d} \in \mathcal{D}}\left(0, h_t(\mathbf{d}) - h_m^{\max}\right) \mid \mathbf{d}^{(m+1)} = \mathbf{d}, \mathbf{D}_m, \mathbf{h}_m\right] \\
&= \left(\bar{\mathcal{H}}_t(\mathbf{d}) - h_m^{\max}\right) \Phi\left(\left(\mu_t(\mathbf{d}) - h_m^{\max}\right) / \sqrt{\text{cov}_t(\mathbf{d}, \mathbf{d})}\right) \\
&\quad + \sqrt{\text{cov}_t\left(\mathbf{d}, \mathbf{d}\right)}\, \phi\left(\left(\mu_t(\mathbf{d}) - h_m^{\max}\right) / \sqrt{\text{cov}_t(\mathbf{d}, \mathbf{d})}\right),
\end{aligned} \tag{15}$$

where $\Phi(.)$ and $\phi(.)$ are the cumulative standard normal and the standard normal density, and the expectation in (15) is with respect to the posterior distribution at iteration m. Notice that the expected improvement policy measures gain $(h_m^{\max} - h_t(\mathbf{d}))$ if $h_t(\mathbf{d})$ turns out to be larger than $h_m^{\max}$, and no gain otherwise. One can see in (15) that there is a closed-form solution for computation of the expected improvement, i.e., $\alpha^t_{\mathrm{EI}}(\mathbf{d})$. It should be noted that other Bayesian optimization acquisition functions, such as knowledge gradient [48] and entropy search [49], can be used instead of the expected improvement policy [29], [50] in selection processes.

It is worth mentioning that optimization techniques based on non-GP surrogate models, also referred to as sequential model-based optimization algorithms [51], [52], can also be used for the selection process in non-continuous design spaces. However, the main advantage of using the graph-based kernel function in the Gaussian process surrogate model in this paper is the possibility of employing arbitrary Bayesian optimization techniques for an efficient experimental design process.

For very large design spaces, finding the exact solution for maximization in (14) could be computationally expensive. This comes from the need for evaluating the acquisition function, i.e., α^t_{EI}, at all nodes of the graph (i.e., all samples in the design space). The following two ways can be considered for overcoming the aforementioned difficulties:

1) Employing a population-based optimization technique such as particle swarm optimization technique [53], genetic algorithm [54], [55], the breadth-first local search (BFLS) [56] or any other optimization technique.
2) Performing a greedy search over the vertices of the graph. This can be done by selecting a set of experiments (i.e., nodes of the graph) and comparing their acquisition functions with their neighboring vertices. Then, one needs to move in a greedy way to the vertices with the higher acquisition function values. The procedure should continue until reaching a fixed number of steps or when no movement in all vertices is possible.

After the selection of $\mathbf{d}^{(m+1)}$ using (14), the MOCU-Reduction function can be approximated for the design $\mathbf{d}^{(m+1)}$. Then, the Gaussian process should be updated according to all available information, denoted by $\mathbf{D}_{m+1} = (\mathbf{D}_m, \mathbf{d}^{(m+1)})$ and $\mathbf{h}_{m+1} = [\mathbf{h}_m, h(\mathbf{d}^{(m+1)})]^T$. The process can stop when reaching a fixed number of iterations, or when the changes in the maximum of the mean of GP in consecutive iterations falls below a small pre-specified threshed.

Assuming the process stops at iteration M, the next design to be conducted can be selected according to the mean of the latest constructed GP given all available information as:

$$\begin{aligned} \mathbf{d}_{t+1} &= \operatorname*{argmax}_{\mathbf{d}\in\mathcal{D}} \bar{\mathcal{H}}_t(\mathbf{d}) \\ &= \operatorname*{argmax}_{\mathbf{d}\in\mathcal{D}} \mathbb{E}[\mathcal{H}_t(\mathbf{d}) \mid \mathbf{D}_M, \mathbf{h}_M]. \end{aligned} \tag{16}$$

Upon selection of $\mathbf{d}_{t+1}$, the experiment should be conducted over $\mathbf{d}_{t+1}$ and the outcome of the experiment y_{t+1} should be used for updating the posterior distribution over the unknwon parameter (i.e., $p(\boldsymbol{\theta} \mid \mathbf{d}_{1:t+1}, \mathbf{y}_{1:t+1})$). This updated posterior then can be used for computation of the new MOCU-Reduction function, $h_{t+1}(.)$, and selection of the next experiment.

The algorithm and the schematic diagram of the proposed framework are presented in Algorithm 1 and Figure 1 respectively.

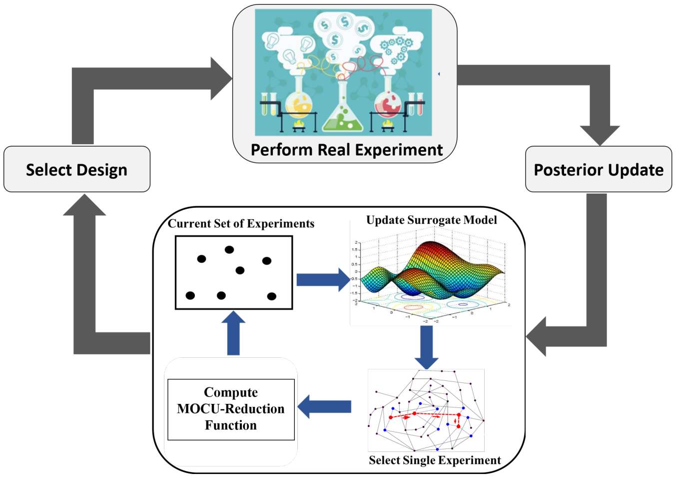

Fig. 1: Schematic diagram of the proposed GBO-MOCU framework.

IV. EXPERIMENTAL DESIGN FOR STRUCTURAL INTERVENTION IN GENOMICS

In genomic signal processing, the main goal is to derive intervention strategies for gene regulatory networks (GRNs) to shift the systems from undesirable states such as carcinogenic states [20], [57]–[59] to desirable ones. In the following paragraphs, the Boolean network model, the structural intervention, and experimental design in the GRNs context are described.

A. Boolean Network Model of Gene Regulatory Networks (GRNs)

Several models have been developed to capture the dynamical behavior of gene regulatory networks. These include the ordinary differential equation (ODE) models [60], [61], Bayesian network models [62]–[65] and Boolean network models [33], [66]–[69]. Unlike most models which suffer from overfitting problem, large number of parameters and lack of interpretability, the Boolean network models can effectively model the dynamical processes, as well as discover causal interactions.

The Boolean network with perturbation (BNp) [57] is a well-known Boolean network model capable of capturing the dynamical behavior of gene regulatory networks. The state values of a gene regulatory network with n genes at time step k can be described by $\{\mathbf{X}(t); k = 0, 1, \ldots\}$, where $\mathbf{X}(t) = [\mathbf{X}_1(t), ..., \mathbf{X}_n(t)]^T$ is a vector of size n. The ith gene at any time k takes a real value from the set $\{0, 1\}$, where 0 and 1 correspond to cases that the ith gene is OFF and ON, respectively. The state values of genes are updated through the following process:

$$\mathbf{X}(t+1) = \mathbf{F}\left(\mathbf{X}(t)\right) \oplus \boldsymbol{\eta}(t), \tag{17}$$

for $k = 0, 1, \ldots$; where $\oplus$ is component-wise modulo 2 addition, $\boldsymbol{\eta}(t)$ is the transition noise at time k, and $\mathbf{F}$ is the *network function*. The noise process $\boldsymbol{\eta}(t) = [\eta_1(t), \eta_2(t), \cdots, \eta_n(t)]$ is assumed to have independent components distributed as $\eta_i(t) \sim \mathrm{Bernoulli}(p)$, where $p > 0$ denotes the amount of "perturbation" to the Boolean state process.

Algorithm 1 The Proposed Graph-Based Bayesian Optimization Mean Objective Cost of Uncertainty (GBO-MOCU)

```
Initialization: Set the following values: $T_{\text{Max}}$ : Total Number of Experiments; $\epsilon$ : Stopping Criterion Threshold; $P(\boldsymbol{\theta})$: Initial Distribution.
for $t = 0$ to $T_{\text{Max}} - 1$ do
    Surrogate Model Construction: Initialize the hyper-parameters of the Gaussian process $\mathcal{GP}(0, k_t(.,.))$, i.e., $\beta_t, \sigma_t$.
    $\mathbf{D}_0 = \{\}$, $\mathbf{h}_0 = \{\}$, $m = 0$, flag = 0.
    while flag = 0 do
        Acquisition Function Computation: Evaluate acquisition function, $\alpha^t_{\text{EI}}$, on vertices in $\mathcal{D}$ using Eq. (15).
        Design Selection: Select the best design input: $\mathbf{d}^{(m+1)} = \text{argmax}_{\mathbf{d} \in \mathcal{D}}\, \alpha^t_{\text{EI}}(\mathbf{d})$.
        MOCU Computation: Compute $h_t(\mathbf{d}^{(m+1)})$ using (5).
        Training Data Augmentation: Update the GP training data: $\mathbf{D}_{m+1} = \{\mathbf{D}_m, \mathbf{d}^{(m+1)}\}, \mathbf{h}_{m+1} = \{\mathbf{h}_m, h_t(\mathbf{d}^{(m+1)})\}$.
        Surrogate Model Update: Update the hyper-parameters of the GP, i.e., $\beta_t$, $\sigma_t$.
        If $|\max_{\mathbf{d} \in \mathcal{D}} \bar{\mathcal{H}}_m(\mathbf{d}) - \max_{\mathbf{d}' \in \mathcal{D}} \bar{\mathcal{H}}_{m+1}(\mathbf{d}')| < \epsilon$, set flag = 1.
        $m = m + 1$;
    end while
    Real Experiment: Select $\mathbf{d}_{t+1} = \text{argmax}_{\mathbf{d} \in \mathcal{D}}\, \bar{\mathcal{H}}_m(\mathbf{d})$ and conduct the experiment over $\mathbf{d}_{t+1}$ to get the outcome $y_{t+1}$.
    Posterior Update: Update the posterior distribution: $P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t+1}, y_{1:t+1})$.
end for
```

The Boolean function representing the temporal dynamics of gene regulatory networks with n genes can be represented as $\mathbf{F} = (f_1, ..., f_n)$. A commonly used network model for capturing the dynamical behavior of gene regulatory networks is based on the majority voting rule [70]–[73] and can be described as:

$$f_i\big(\mathbf{X}(t)\big) = \begin{cases} 1 & \text{If } \sum_j R_{ij}\, X_j(t) > 0, \\ 0 & \text{If } \sum_j R_{ij}\, X_j(t) < 0, \\ X_i(t) & \text{If } \sum_j R_{ij}\, X_j(t) = 0, \end{cases} \tag{18}$$

for $i = 1, ..., n$, where $R_{ij} = +1$ and $R_{ij} = -1$ indicate an activating regulation ($\mathcal{A}$) and a suppressive regulation ($\mathcal{S}$) from gene j to gene i respectively, and $R_{ij} = 0$ represents the case when gene j is not an input to gene i.

Let $\{\mathbf{x}^1, ..., \mathbf{x}^{2^n}\}$ be a set of all possible state vectors. The controlled transition matrix or the transition probability matrix (TPM) is indicated by $\mathbf{P}$ and can be computed as:

$$\begin{aligned} p_{ij} &= P\Big(\mathbf{X}(t+1) = \mathbf{x}^i \mid \mathbf{X}(t) = \mathbf{x}^j\Big) \\ &= p^{||\mathbf{x}^i \oplus \mathbf{F}(\mathbf{x}^j)||_1}(1-p)^{n - ||\mathbf{x}^i \oplus \mathbf{F}(\mathbf{x}^j)||_1}, \end{aligned} \tag{19}$$

for $i, j = 1, ..., 2^n$, where the ith row and jth column of the TPM matrix $\mathbf{P}$ is indicated by p_{ij}, and $||.||_1$ is the L_1 norm.

The steady-state distribution (SSD) of the gene regulatory network modeled by the TPM in (19) can be computed as $\pi^T = \pi^T \mathbf{P}$, where π^T is the transpose of π. .

B. Structural Intervention of GRNs

Several intervention strategies have been developed in recent years for various Boolean network models [20], [21], [57], [74], [75]. These techniques can be grouped into two main categories: *Dynamical Intervention* [21], [33], [59], [76], [77] and *Structural Intervention* [20], [34], [74]. Dynamical intervention aims to make temporal changes in networks to reduce the probability of observing undesirable states either in a long run (i.e., infinite-horizon) or a fixed horizon (i.e., finite-horizon) [21], [33], [38], [77], [78]. The structural intervention, however, aims to make a single change in network structure of GRNS to beneficially change the steady-state distribution of networks. The optimal solution for this type of intervention has been provided in [20] and is described in the following paragraphs.

The rank-1 function perturbation is a well-known type of structural intervention. Let $\{\mathbf{x}^1, \mathbf{x}^2, ...\mathbf{x}^{2^n}\}$ be 2^n different states for a GRN with n genes. According to this intervention, the system transition probability matrix (TPM) $\mathbf{P}$ is altered to $\tilde{\mathbf{P}} = \mathbf{P} + \mathbf{c}\mathbf{b}^T$ [20], where $\mathbf{c}$ and $\mathbf{b}$ are arbitrary vectors and $\mathbf{b}^T\mathbf{e} = 0$ ($\mathbf{e}$ is an all unity column vector). In this paper, we consider a single-gene perturbation process, which is a special case of a rank-1 perturbation. This intervention only changes the output state of a single input state, while the outputs of other states stay unchanged. Let $\mathbb{A}$ be the rank-1 perturbation space, and $\tilde{\mathbf{F}} = (\tilde{f}_1, ..., \tilde{f}_n)$ be the Boolean functions after intervention. Let also consider the structural intervention taken over (j, s). This intervention changes the output value of the Boolean function $\mathbf{x}^s = \tilde{\mathbf{F}}(\mathbf{x}^j) \neq \mathbf{F}(\mathbf{x}^j) = \mathbf{x}^r$, and $\tilde{\mathbf{F}}(\mathbf{x}^i) = \mathbf{F}(\mathbf{x}^i)$, for $i = 1, ..., 2^n$ and $i \neq j$.

The TPM after the perturbation over (j,s) can be represented by $\tilde{\mathbf{P}}$, where the elements of the altered TPM are $\tilde{p}_{jr} = p_{jr} - (1-p)^n$ and $\tilde{p}_{js} = p_{js} + (1-p)^n$. Letting π and $\tilde{\pi}$ be the steady state distributions of the system before and after intervention, the new steady state distribution impacted by the changes in the TPM can be computed as [20]:

$$\tilde{\pi}_i(j,s) = \pi_i + \frac{(1-p)^n \, \pi_j \, (z_{si} - z_{ri})}{1 - (1-p)^n \, (z_{sj} - z_{rj})}\,, \tag{20}$$

where $\mathbf{Z} = [\mathbf{I} - \mathbf{P} + \mathbf{e}\pi^T]^{-1}$ is the fundamental matrix, $\mathbf{I}$ is the diagonal identity matrix of size n, and $z_{si}, z_{ri}, z_{sj}, z_{rj}$ are elements of $\mathbf{Z}$.

Let $\tilde{\pi}_U(j,s) = \sum_{i \in U} \tilde{\pi}_i(j,s)$ be the steady-state probability mass of undesirable states after applying a (j,s) intervention, where U is the set of undesirable Boolean states. The optimal structural intervention can be computed through the following minimization problem:

$$(j^*, s^*) = \underset{j,s \in \{1,2,\ldots,2^n\}}{\operatorname{argmin}} \; \tilde{\pi}_U(j,s)\,. \tag{21}$$

C. Experimental Design in GRNs' Intervention

The complexity in most GRNs poses a huge amount of uncertainty in the modeling process. Most of these uncertainties appear in terms of unknown interactions between different genes in the network. Consider a GRN with L unknown regulations, denoted by $r^1, r^2, \ldots, r^L$, using the model presented in (18), each non-zero interaction could be either $\mathcal{A}$ or $\mathcal{S}$. This means that L unknown interactions result in 2^L different possible network models represented by $\Theta = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{2^L}\}$, where $\boldsymbol{\theta}_j \in \{\mathcal{A}, \mathcal{S}\}^L$, for $j = 1, \ldots, 2^L$. Let also $P(\boldsymbol{\theta})$ be the prior distribution over the network models defined as $P(\boldsymbol{\theta}) = P\big(r^1 = \boldsymbol{\theta}(1), r^2 = \boldsymbol{\theta}(2), \cdots, r^L = \boldsymbol{\theta}(L)\big)$, for $\boldsymbol{\theta} \in \Theta$. There exist L experiments $\mathcal{D} = \{\mathbf{d}_1, \ldots., \mathbf{d}_L\}$, where $\mathbf{d}_i$ determines the regulation r^i. This type of experimentation is common in genomics studies, where each experiment can completely reveal the true type of targeted interaction.

Let $\xi_{\boldsymbol{\theta}}(\mathbf{a})$ be the cost representing the steady-state mass of undesirable states after performing the intervention $\mathbf{a} \in \mathbb{A}$ for the network denoted by the uncertainty $\boldsymbol{\theta} \in \Theta$. The optimal intervention associated with the whole uncertainty in the network is $\mathbf{a}_{\boldsymbol{\theta}} = (j^*_{\boldsymbol{\theta}}, s^*_{\boldsymbol{\theta}})$, where $\xi_{\boldsymbol{\theta}}(\mathbf{a}) \geq \xi_{\boldsymbol{\theta}}(\mathbf{a}_{\boldsymbol{\theta}})$ for any $\mathbf{a} \in \mathbb{A}$.

The MOCU relative to an uncertainty class represented by taken experiments $\mathbf{d}_{1:t} = (\mathbf{d}_1, \ldots, \mathbf{d}_t)$ and the outcomes $y_{1:t} = (y_1, \ldots, y_t)$ for a class of interventions $\mathbb{A}$ can be defined by:

$$\begin{aligned} M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}} &= \mathbb{E}_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}} \left[\xi_{\boldsymbol{\theta}}(\mathbf{a}_{\mathrm{OB}}) - \xi_{\boldsymbol{\theta}}(\mathbf{a}_{\boldsymbol{\theta}})\right] \\ &= \sum_{i=1}^{2^L} P(\boldsymbol{\theta}_i \mid \mathbf{d}_{1:t}, y_{1:t}) \left[\xi_{\boldsymbol{\theta}_i}(\mathbf{a}_{\mathrm{OB}}) - \xi_{\boldsymbol{\theta}}(\mathbf{a}_{\boldsymbol{\theta}_i})\right]\,, \end{aligned} \tag{22}$$

where $\mathbf{a}_{\mathrm{OB}}$ is an optimal Bayesian (OB) intervention defined in (2).

The goal of sequential MOCU-based experimental design in structural intervention in GRNs is to select a regulation (i.e., experiment) at each step that results in the maximal reduction in MOCU in the next step:

$$\begin{aligned} \mathbf{d}_{t+1} &= \underset{\mathbf{d}_j \in \{\mathbf{d}_1, \ldots., \mathbf{d}_L\}}{\operatorname{argmax}} \; \mathbb{E}_t\Big[M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}} - M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}, \mathbf{d}_{t+1} = \mathbf{d}_j}\Big] \\ &= \underset{\mathbf{d}_j \in \{\mathbf{d}_1, \ldots., \mathbf{d}_L\}}{\operatorname{argmax}} \Bigg[M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}} - \Bigg(p_t^j \, M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}, \mathbf{d}_{t+1} = \mathbf{d}_j, y_{t+1} = \mathcal{A}} + \\ &\qquad (1 - p_t^j)\, M_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}, \mathbf{d}_{t+1} = \mathbf{d}_j, y_{t+1} = \mathcal{S}}\Bigg)\Bigg], \end{aligned} \tag{23}$$

where p_t^j is the probability that the jth regulation is activating ($\mathcal{A}$) according to the posterior distribution $P(\boldsymbol{\theta} \mid \mathbf{d}_{1:t}, y_{1:t})$, which can be computed as:

$$\begin{aligned} p_t^j &= P(r^j = \mathcal{A} \mid \mathbf{d}_{1:t}, y_{1:t}) = \mathbb{E}_{\boldsymbol{\theta}|\mathbf{d}_{1:t}, y_{1:t}}[1_{\boldsymbol{\theta}(j) = \mathcal{A}}] \\ &= \sum_{j=1}^{2^L} P\big(\boldsymbol{\theta}(j) \mid \mathbf{d}_{1:t}, y_{1:t}\big)\, 1_{\boldsymbol{\theta}(j) = \mathcal{A}}, \end{aligned} \tag{24}$$

and $1_{\boldsymbol{\theta}(j) = \mathcal{A}}$ if $\boldsymbol{\theta}(j) = \mathcal{A}$ and 0 otherwise.

After taking the experiment $\mathbf{d}_{t+1}$ and getting outcome y_{t+1}, one needs to update the posterior probability over the Boolean network models and repeat the same process for selecting the next experiment. However, a large number of unknown regulations and a huge cost associated with each experiment limit the total number of experiments that can be taken. This magnifies the need for an experimental design process for taking the minimum number of experiments helping the most during the structural intervention process.

V. NUMERICAL EXPERIMENTS

A. Simulation Set-Up

The prior probability defined over network models depends on the available prior knowledge. For instance, in gene regulatory networks, described in Section IV, the prior probability represents the biologist/expert knowledge regarding the type of interaction between genes in regulatory networks. For large and less-studied regulatory networks with no or limited available prior knowledge, the prior distribution is similar to the uniform distribution. In contrast, for small and well-known regulatory networks, the prior probability is peaked over some models in parameter space. To investigate the proposed framework's performance in all ranges of proper distributions, the simulations are conducted for various prior distributions over the network's uncertainty. Toward this, we consider the symmetric Dirichlet distribution for prior distribution as:

$$P(.) \sim f(P(.);\phi) = \frac{\Gamma\left(\phi 2^L\right)}{\Gamma(\phi)^{2^L}} \prod_{j=1}^{2^L} P(\boldsymbol{\theta}_j)^{\phi - 1}, \tag{25}$$

where Γ is the gamma function and $\phi > 0$ is the symmetric Dirichlet distribution parameter. This distribution is capable of generating a wide range of prior distributions: the small values of ϕ generate prior distributions that are highly peaked, whereas the large values of ϕ correspond to the prior distributions closer to the uniform.

The following experimental design strategies are considered in the numerical results: 1) The proposed GBO-MOCU framework, 2) Best-First Search [56], 3) Entropy Experimental Design Technique [13], [79], 4) Sequential Model-based Algorithm Configuration (SMAC) [51], and 5) Random Search. For all numerical results, we represent the Bayesian intervention cost over the true network model, where the Bayesian intervention is computed according to the posterior distribution over the network models.

SMAC [51] is a popular model-based optimization technique. This method constructs a regression tree to predict the function values and uses this model to select the next design sample. The Best-First search selects 10% of the total number of design samples randomly. Then, 20 design samples with the highest function values are used as initial points for optimization. The search then continues by a local search over the design samples close to the selected samples. The process is repeated until no function value on adjacent samples is higher than the current design sample's function value. Best-First search is a local search; however, the initial random search and multi-starts are used to escape from local optima.

The results of the proposed framework are compared with the entropy experimental design policy [13], [79]. The entropy policy only accounts for the uncertainty, as opposed to the objective, in the selection process. The entropy for the posterior distribution $pt = [P(\boldsymbol{\theta}_1 \mid \mathbf{d}_{1:t}, y_{1:t}), \ldots, P(\boldsymbol{\theta}_{2^L} \mid \mathbf{d}_{1:t}, y_{1:t})]^T$ can be computed as:

$$H_{\boldsymbol{\theta}|\mathbf{d}_{1:t},y_{1:t}} = -\sum_{j=1}^{2^L} pt(j)\, \log_2 pt(j), \tag{26}$$

where the entropy takes a value between 0 and L. The values of entropy close to L indicate the case when all network functions have almost equal posterior values (i.e., maximum uncertainty).

The Entropy approach sequentially chooses an experiment which maximizes the reduction in expected entropy at the next step:

$$\begin{aligned}\mathbf{d}_{t+1} &= \underset{\mathbf{d}\in\{\mathbf{d}_1,\ldots,\mathbf{d}_L\}}{\operatorname{argmax}} \;\; \mathbb{E}_t\left[H_{\boldsymbol{\theta}|\mathbf{d}_{1:t},y_{1:t}} - H_{\boldsymbol{\theta}|\mathbf{d}_{1:t},y_{1:t},\mathbf{d}_{t+1}=d}\right] \\ &= \underset{\mathbf{d}\in\{\mathbf{d}_1,\ldots,\mathbf{d}_L\}}{\operatorname{argmax}} \; -\Bigg(p_t^j\, H_{\boldsymbol{\theta}|\mathbf{d}_{1:t},y_{1:t},\mathbf{d}_{t+1}=\mathbf{d}_j,y_{t+1}=\mathcal{A}} + \\ &\qquad (1-p_t^j)\, M_{\boldsymbol{\theta}|\mathbf{d}_{1:t},y_{1:t},\mathbf{d}_{t+1}=\mathbf{d}_j,y_{t+1}=\mathcal{S}}\Bigg),\end{aligned} \tag{27}$$

where p_t^j is defined in (24).

B. Synthetic BNps

The synthetic GRNs are used here for evaluating the performance of the proposed framework. According to the model proposed in the previous section, we generate 500 random GRNs with 20 genes, in which each network has L unknown regulations. Interactions between the components of random BNs are selected randomly from $\{-1, 0, +1\}$ with the constraint that each component has at most two non-zero inputs. The Bernoulli parameter, specifying the stochasticity in the network model, is assumed to be $p = 0.001$. For the intervention purpose, we assume observing the activation of the first gene undesirable, denoting the undesirable set $U = \{\mathbf{x}^1, \ldots, \mathbf{x}^{524,288}\}$. All numerical experiments are based on the fixed set of values for the system parameters displayed in Table I.

TABLE I: Parameter values for Synthetic BNps.

Parameter	**Value**
Total Number of Experiments, T_{Max}	1 to 150
Stopping Criterion Threshold, ϵ	0.02
Initial Values for β_t	Uniform $[0, 2]$
Initial Values for σ_t	Uniform $[0, 2]$
Dirichlet Distribution Parameter, ϕ	$0.1, 1, 100$
Number of Genes, n	20
Number of Unknown Regulations, L	$60, 100, 150$
Process Noise, p	0.001
Undesirable States	$U = \{\mathbf{x}^1, \ldots, \mathbf{x}^{524,288}\}$

20-gene random Boolean networks with 60 and 100 unknown regulations (L) are considered in this part of numerical experiments. The average cost of Bayesian intervention with respect to the number of conducted experiments for different experimental design strategies is shown in Figure 2 and 3. It can be seen that the system under the proposed GBO-MOCU experimental design policy has achieved the lowest average cost of Bayesian intervention in comparison to other techniques. This comes from the fact that the experiment selected by the GBO-MOCU policy at each step is based on all the previously taken experiments and their outcomes during the selection process. By contrast, the Best-First and Random policies do not model the correlation over the experimental space, and as a result, systems under these policies undergo a much higher cost of Bayesian intervention.

Comparing the results for different prior distributions (ϕ), one can see that the initial average cost of intervention is larger for more uniform prior distributions (larger ϕ). This comes from the fact that the Bayesian intervention becomes less accurate in the presence of less-peaked (more uniform) distributions. The MOCU-based strategies achieve the maximum reduction in cost for $\phi = 0.1$, which is due to the fact that having a few informative experiments in the presence of rich prior distribution results in a significant reduction in the intervention cost.

The number of unknown regulations in the numerical results provided in Figure 2 and 3 is 60 and 100, respectively. As expected, the amount of intervention cost is larger in the presence of larger unknown regulations, and a much larger number of experiments have been taken to achieve the same intervention performance in cases with larger unknown regulations. An instance of that can be seen by comparing the proposed framework's performance in the left plots of Figures 2 and 3. In fact, the cost value of 0.1 is achieved after taking 7 experiments for the case with 60 unknown regulations, whereas the same average cost (i.e., 0.1) is achieved after taking 27 experiments for the case with 100 unknown regulations.

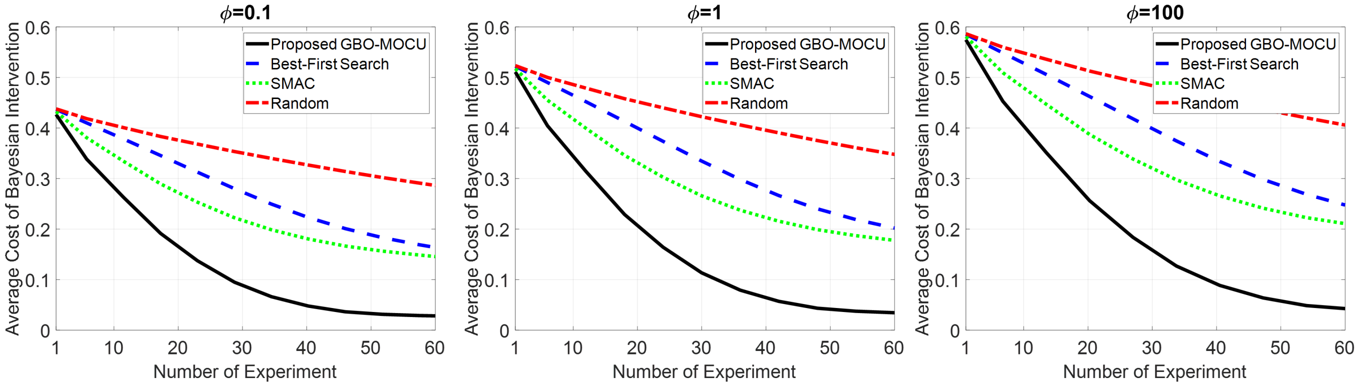

Fig. 2: The average cost of Bayesian intervention for synthetic Boolean networks. The results are averaged over 500 randomly generated networks with 60 unknown regulations (L).

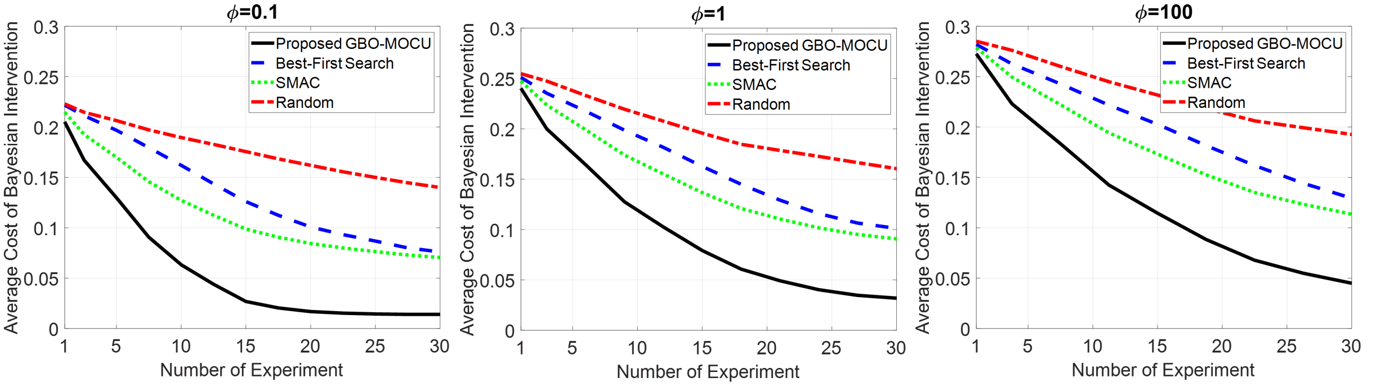

Fig. 3: The average cost of Bayesian intervention for synthetic Boolean networks. The results are averaged over 500 randomly generated networks with 100 unknown regulations (L).

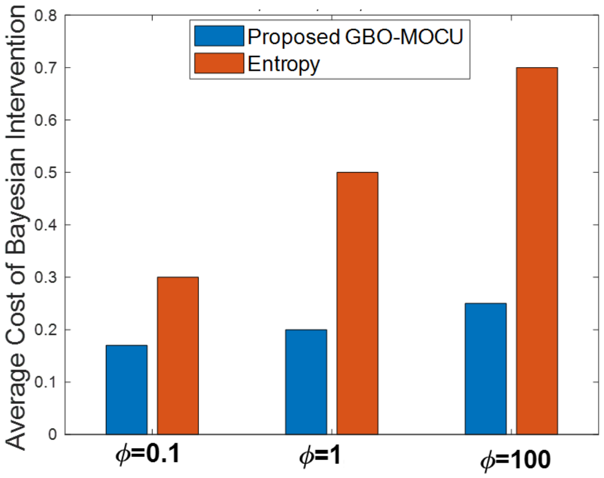

Fig. 4: The average cost of Bayesian intervention for the proposed GBO-MOCU framework and the Entropy policy for synthetic Boolean networks. The results are averaged over 500 randomly generated networks with 40 unknown regulations (L).

In this part of the experiments, the results of the proposed GBO-MOCU policy and the Entropy policy are compared for 20-gene random Boolean networks with L = 40 unknown regulations. The results after taking 10 experiments are shown in Figure 4. One can see that the average cost of Bayesian intervention is lower for more non-uniform distributions (i.e., smaller ϕ). However, as the prior distribution gets closer to a uniform one (larger ϕ), much higher intervention costs are obtained for systems under the Entropy policy. Indeed, the Entropy scheme does not discriminate between potential

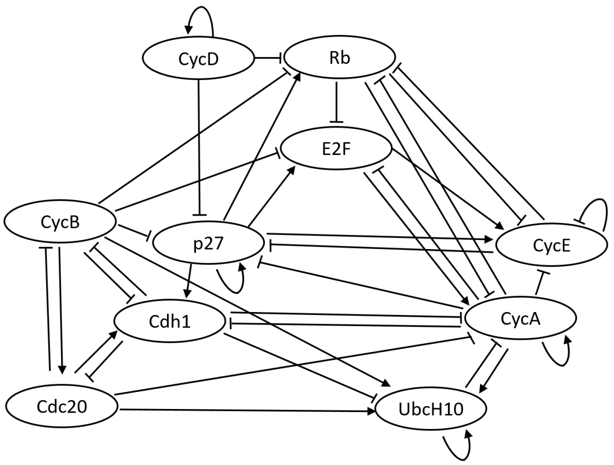

Fig. 5: Pathway diagram for the cell-cycle mammalian network. The normal and blunt arrows denote activating and suppressive regulations respectively.

experiments and performs like a random policy for less-peaked prior distributions. By contrast, the GBO-MOCU policy is capable of selecting the most effective experiments to reduce the intervention cost (as opposed to only uncertainty). Thus, systems under the Entropy policy have undergone a much higher average intervention cost in comparison to the GBO-MOCU policy in cases with more uniform prior distributions. Meanwhile, comparing the results of the proposed framework and Best-First Search and SMAC, one can see the concrete performance of the proposed framework. The Best-First Search's low performance comes from the random selection of the design samples and local search around them. The SMAC technique requires a huge number of early function evaluations for reliable construction of the deterministic surrogate model (i.e., regression tree), leading to the optimization process's inefficiency. However, the Bayesian surrogate model in the proposed framework allows predicting the distribution of the MOCU values and selecting the next sample by considering all potential outcomes instead of a single predicted value.

In this part of the experiment, for 500 randomly generated networks with 150 unknown regulations (L), the average cost of Bayesian intervention is shown in Figure 6. The total number of experiments is varied between 1 and 150, where the uncertainty in any selected interaction disappears after taking each experiment. Therefore, as expected, the average cost of intervention is the same when 150 experiments are taken for all methods due to removing all uncertainties in the network interactions. However, in practice, the number of experiments to be taken is often limited due to the real cost of conducting experiments. Thus, looking at Figure 6, one can see that for a small number of experiments (i.e., between 1 and 150), the proposed framework significantly outperforms other methods.

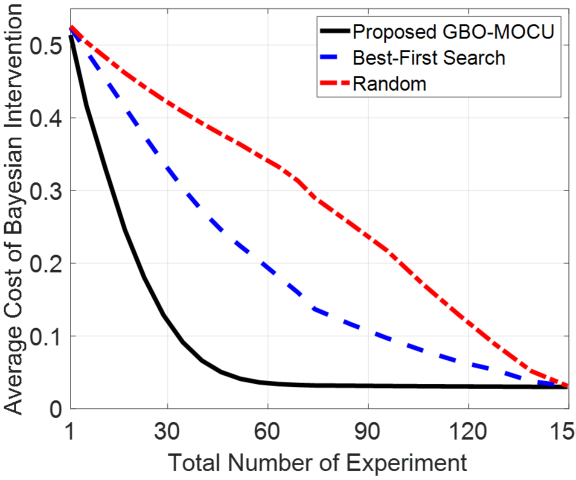

Fig. 6: The average cost of Bayesian intervention with respect to the total number of experiments for synthetic Boolean networks with 150 unknown interactions and $\phi = 1$.

The proposed framework's effectiveness in terms of computational cost compared with the original MOCU policy is demonstrated in this part of the experiments. The experiments have been conducted on a PC with an Intel Core i7-4790 CPU@3.60-GHz clock and 16 GB of RAM, and the running time (per minutes) for both methods is presented in Figure 7. The number of unknown interactions is 150, and the computational time is reported till both experimental design frameworks have performed all 150 experiments. It can be seen that a much faster reduction in the intervention cost is achieved by the proposed framework relative to the original MOCU policy. This is due to the fact that the surrogate model over 150 available experiments significantly helps the proposed framework to select better experiments in comparison to the original MOCU policy, which exhaustively explores all possible experiments for its selection process.

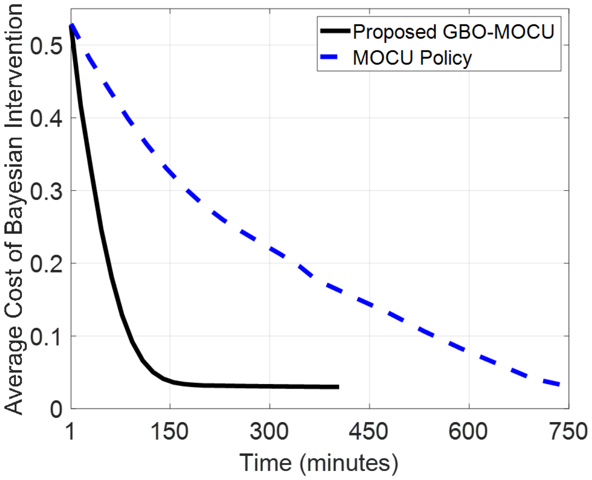

Fig. 7: The average cost of Bayesian intervention with respect to time (per minutes) for the proposed GBO-MOCU framework and the MOCU policy for synthetic Boolean networks with 150 unknown interactions and $\phi = 1$.

C. *Mammalian Cell Cycle Network*

The mammalian cell cycle [68] is considered here for assessing the performance of the proposed framework. Figure 5 represents the pathway diagram for this network, where the blunt and normal arrows represent suppressive and activating regulations, respectively. Mammalian cell division is coordinated with the growth of organisms according to the extracellular signals that control the activation of CycD in the cell. The state vector is $\mathbf{x}$ =(CycD, Rb, p27, E2F, CycE, CycA, Cdc20, Cdh1, UbcH10, CycB). As described in [68], mammalian cell division is tightly controlled. In a growing mammal, the cell division should coordinate with the overall growth of the organism. This coordination is controlled via extracellular signals. These signals indicate whether a cell should divide or remain in a resting state. The positive signals, or growth factors, instigate the activation of CycD in the cell. The key genes in this model are CycD, retinoblastoma (Rb), and p27. Rb is a tumor-suppressor gene. This gene is expressed in the absence of the CycD, which inhibits Rb by phosphorylation. Whenever p27 is present, Rb can be expressed even in the presence of CycE or CycA. Gene p27 is active in the absence of the CycD. Whenever p27 is present, it blocks the action of CycE or CycA and stops the cell cycle.

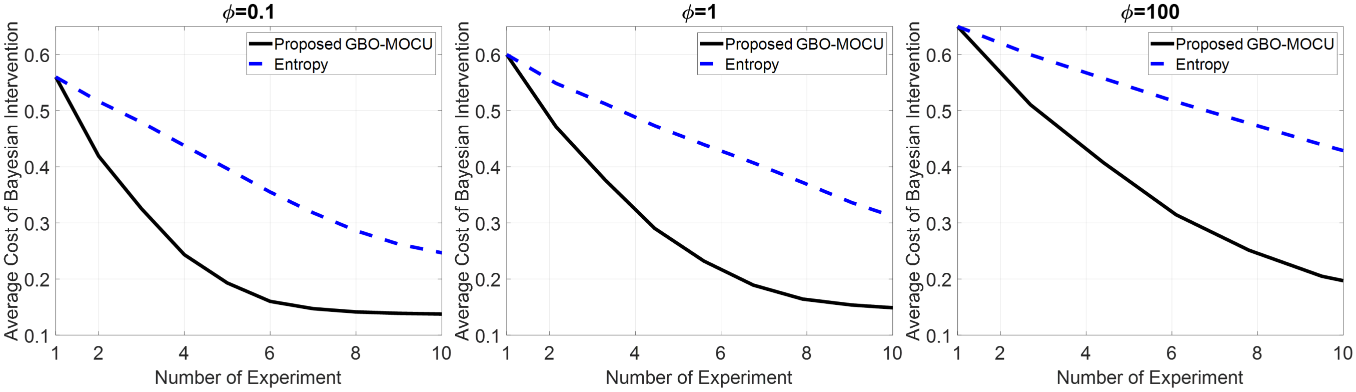

Fig. 8: The average cost of Bayesian intervention with respect to the number of experiments for the Mammalian Cell Cycle network with 20 unknown regulations (L) and three different Dirichlet distributions (ϕ).

Therefore, states with downregulated CycD, Rb, and p27 are undesirable ones, which correspond to cancerous phenotypes and are represented by $U = \{\mathbf{x}^1, ..., \mathbf{x}^{128}\}$. Thus, the goal is to shift the steady-state distribution of undesirable states through structural intervention.

We consider 20 unknown regulations (L) which are selected randomly among all regulations shown in Figure (5). We apply the proposed GBO-MOCU policy and the Entropy policy to predict the experiment to be performed. 500 initial distributions have been generated from three different Dirichlet distributions. The experiments are based on the fixed set of values for the system parameters displayed in Table II.

TABLE II: Parameter values for mammalian cell cycle network.

Parameter	**Value**
Total Number of Experiments, T_{Max}	1 to 10
Stopping Criterion Threshold, ϵ	0.02
Initial Values for β_t	Uniform $[0, 2]$
Initial Values for σ_t	Uniform $[0, 2]$
Dirichlet Distribution Parameter, ϕ	$0.1, 1, 100$
Number of Genes, n	10
Number of Unknown Regulations, L	20
Process Noise, p	0.001
Undesirable States	$U = \{\mathbf{x}^1, ..., \mathbf{x}^{128}\}$

The average intervention cost for the GBO-MOCU policy and the Entropy policy are presented in Figure 8. One can see that the system under the GBO-MOCU policy has achieved the minimum intervention cost after taking a few experiments, whereas the cost reduction is much less significant for the system under the Entropy policy. This again comes from the fact that the proposed framework takes the main objective function into account in the decision-making process. Meanwhile, one can see that the average intervention cost is larger for less-peaked prior distributions (larger ϕ). While the experimental design for both policies becomes more challenging in the presence of less-peaked distributions. The average cost obtained under the Entropy policy is much larger than the proposed framework in those cases. This is due to the fact that the entropy policy behaves like a random policy in the presence of prior distribution.

VI. CONCLUSION

This paper introduced the Graph-Based Bayesian Optimization Mean Objective Cost of Uncertainty (GBO-MOCU) framework for scalable objective-based experimental design. The proposed framework is built on the combination of the graphical model, the Bayesian optimization, and the mean objective cost of uncertainty (MOCU). The difficulties in large discrete or combinatorial design spaces were handled through a graphical model, and the correlation over the MOCU reduction function is considered through the Gaussian process. Using the proposed graph-based Gaussian process, an efficient and closed-form policy for the sequential experimental design was achieved through the well-known expected improvement policy. In numerical experiments using the intervention process of gene regulatory networks, the high performance of the proposed framework is demonstrated.

ACKNOWLEDGMENT

The authors acknowledge the support of the National Science Foundation through the NSF award IIS-1946999.

REFERENCES

[1] J. Drews, "Drug discovery: a historical perspective," *science*, vol. 287, no. 5460, pp. 1960–1964, 2000.
[2] R. Lam, M. Poloczek, P. Frazier, and K. E. Willcox, "Advances in Bayesian optimization with applications in aerospace engineering," in *2018 AIAA Non-Deterministic Approaches Conference*, p. 1656, 2018.
[3] S. F. Ghoreishi and M. Imani, "Bayesian optimization for efficient design of uncertain coupled multidisciplinary systems," in *American Control Conference (ACC)*, IEEE, 2020.
[4] C. F. Ong, J. L. Hicks, and S. L. Delp, "Simulation-based design for wearable robotic systems: an optimization framework for enhancing a standing long jump," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 5, pp. 894–903, 2015.
[5] C. Osorio and L. Chong, "A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems," *Transportation Science*, vol. 49, no. 3, pp. 623–636, 2015.
[6] F. Hamze, Z. Wang, and N. De Freitas, "Self-avoiding random dynamics on integer complex systems," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 23, no. 1, p. 9, 2013.

[7] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Advances in neural information processing systems*, pp. 2546–2554, 2011.

[8] M. Iman and S. F. Ghoreishi, "Scalable inverse reinforcement learning through multi-fidelity Bayesian optimization," *IEEE transactions on neural networks and learning systems*, 2021.

[9] M. Imani, S. F. Ghoreishi, D. Allaire, and U. Braga-Neto, "MFBO-SSM: Multi-fidelity Bayesian optimization for fast inference in state-space models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 7858–7865, 2019.

[10] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Advances in neural information processing systems*, pp. 2951–2959, 2012.

[11] D. C. Montgomery, *Design and analysis of experiments*. John wiley & sons, 2017.

[12] B.-J. Yoon, X. Qian, and E. R. Dougherty, "Quantifying the objective cost of uncertainty in complex dynamical systems," *IEEE Transactions on Signal Processing*, vol. 61, no. 9, pp. 2256–2266, 2013.

[13] D. V. Lindley *et al.*, "On a measure of the information provided by an experiment," *The Annals of Mathematical Statistics*, vol. 27, no. 4, pp. 986–1005, 1956.

[14] T. J. Loredo, "Bayesian adaptive exploration," in *AIP Conference Proceedings*, vol. 707, pp. 330–346, AIP, 2004.

[15] J. van Den Berg, A. Curtis, and J. Trampert, "Optimal nonlinear bayesian experimental design: an application to amplitude versus offset experiments," *Geophysical Journal International*, vol. 155, no. 2, pp. 411–421, 2003.

[16] G. Terejanu, R. R. Upadhyay, and K. Miki, "Bayesian experimental design for the active nitridation of graphite by atomic nitrogen," *Experimental Thermal and Fluid Science*, vol. 36, pp. 178–193, 2012.

[17] M. Iman and S. F. Ghoreishi, "Two-stage bayesian optimization for scalable inference in state space models," *IEEE transactions on neural networks and learning systems*, 2021.

[18] G. Franceschini and S. Macchietto, "Model-based design of experiments for parameter precision: State of the art," *Chemical Engineering Science*, vol. 63, no. 19, pp. 4846–4872, 2008.

[19] J. O. Hollinger and J. C. Kleinschmidt, "The critical size defect as an experimental model to test bone repair materials.," *The Journal of craniofacial surgery*, vol. 1, no. 1, pp. 60–68, 1990.

[20] X. Qian and E. R. Dougherty, "Effect of function perturbation on the steady-state distribution of genetic regulatory networks: Optimal structural intervention," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4966–4976, 2008.

[21] M. Imani and U. M. Braga-Neto, "Control of gene regulatory networks with noisy measurements and uncertain inputs," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 2, pp. 760–769, 2018.

[22] M. Imani and S. F. Ghoreishi, "Bayesian optimization objective-based experimental design," in *American Control Conference (ACC)*, IEEE, 2020.

[23] R. H. Liebeck, "Design of subsonic airfoils for high lift," *Journal of aircraft*, vol. 15, no. 9, pp. 547–561, 1978.

[24] S. F. Ghoreishi and M. Iman, "Bayesian surrogate learning for uncertainty analysis of coupled multidisciplinary systems," *Journal of Computing and Information Science in Engineering*, 2021.

[25] P. Sebastiani and H. P. Wynn, "Maximum entropy sampling and optimal Bayesian experimental design," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 1, pp. 145–157, 2000.

[26] T. Lundstedt, E. Seifert, L. Abramo, B. Thelin, Å. Nyström, J. Pettersen, and R. Bergman, "Experimental design and optimization," *Chemometrics and intelligent laboratory systems*, vol. 42, no. 1-2, pp. 3–40, 1998.

[27] R. Dehghannasiri, B.-J. Yoon, and E. R. Dougherty, "Optimal experimental design for gene regulatory networks in the presence of uncertainty," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 12, no. 4, pp. 938–950, 2015.

[28] M. Imani, R. Dehghannasiri, U. M. Braga-Neto, and E. R. Dougherty, "Sequential experimental design for optimal structural intervention in gene regulatory networks based on the mean objective cost of uncertainty," *Cancer informatics*, vol. 17, 2018.

[29] J. Mockus, V. Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," *Towards global optimization*, vol. 2, no. 117-129, p. 2, 1978.

[30] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.

[31] E. R. Dougherty, R. Pal, X. Qian, M. L. Bittner, and A. Datta, "Stationary and structural control in gene regulatory networks: basic concepts," *International Journal of Systems Science*, vol. 41, no. 1, pp. 5–16, 2010.

[32] M. Iman and S. F. Ghoreishi, "Partially-observed discrete dynamical systems," in *Proceedings of the 2021 American Control Conference (ACC 2021)*, IEEE, 2021.

[33] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, no. 2, pp. 261–274, 2002.

[34] I. Shmulevich, E. R. Dougherty, and W. Zhang, "Control of stationary behavior in probabilistic Boolean networks by means of structural intervention," *Journal of Biological Systems*, vol. 10, no. 04, pp. 431–445, 2002.

[35] J. P. Holman and W. J. Gajda, *Experimental methods for engineers*, vol. 2. McGraw-Hill New York, 2001.

[36] D. J. MacKay, "Information-based objective functions for active data selection," *Neural computation*, vol. 4, no. 4, pp. 590–604, 1992.

[37] X. Huan and Y. M. Marzouk, "Sequential Bayesian optimal experimental design via approximate dynamic programming," *arXiv preprint arXiv:1604.08320*, 2016.

[38] M. Imani, S. F. Ghoreishi, and U. M. Braga-Neto, "Bayesian control of large MDPs with unknown dynamics in data-poor environments," in *Advances in neural information processing systems*, pp. 8156–8166, 2018.

[39] M. Imani, E. Dougherty, and U. Braga-Neto, "Boolean Kalman filter and smoother under model uncertainty," *Automatica*, vol. 111, p. 108609, 2020.

[40] X. Cai, J. A. Bazerque, and G. B. Giannakis, "Inference of gene regulatory networks with sparse structural equation models exploiting genetic perturbations," *PLoS computational biology*, vol. 9, no. 5, 2013.

[41] S. M. Ud-Dean and R. Gunawan, "Optimal design of gene knockout experiments for gene regulatory network inference," *Bioinformatics*, vol. 32, no. 6, pp. 875–883, 2015.

[42] M. Imani and S. F. Ghoreishi, "Optimal finite-horizon perturbation policy for inference of gene regulatory networks," in *IEEE Intelligent Systems*, IEEE, 2020.

[43] T. E. Ideker, V. THORSSONt, and R. M. Karp, "Discovery of regulatory interactions through perturbation: inference and experimental design," in *Biocomputing 2000*, pp. 305–316, World Scientific, 1999.

[44] C. Oh, J. Tomczak, E. Gavves, and M. Welling, "Combinatorial Bayesian optimization using the Graph Cartesian Product," in *Advances in Neural Information Processing Systems*, pp. 2910–2920, 2019.

[45] A. Ortega, P. Frossard, J. Kovavcević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[46] C. E. Rasmussen and C. Williams, *Gaussian processes for machine learning*. MIT Press, 2006.

[47] A. Doucet, M. Pitt, and R. Kohn, "Efficient implementation of MCMC when using an unbiased likelihood estimator," tech. rep., Working Paper, 2012.

[48] P. Frazier, W. Powell, and S. Dayanik, "The knowledge-gradient policy for correlated normal beliefs," *INFORMS journal on Computing*, vol. 21, no. 4, pp. 599–613, 2009.

[49] J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani, "Predictive entropy search for efficient global optimization of black-box functions," in *Advances in neural information processing systems*, pp. 918–926, 2014.

[50] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *Journal of global optimization*, vol. 21, no. 4, pp. 345–383, 2001.

[51] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *International conference on learning and intelligent optimization*, pp. 507–523, Springer, 2011.

[52] F. Hutter, H. Hoos, and K. Leyton-Brown, "An evaluation of sequential model-based optimization for expensive blackbox functions," in *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, pp. 1209–1216, 2013.

[53] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4, pp. 1942–1948, IEEE, 1995.

[54] E. J. Anderson and M. C. Ferris, "Genetic algorithms for combinatorial optimization: the assemble line balancing problem," *ORSA Journal on Computing*, vol. 6, no. 2, pp. 161–173, 1994.

[55] D. Whitley, "A genetic algorithm tutorial," *Statistics and computing*, vol. 4, no. 2, pp. 65–85, 1994.

[56] H. Atabakhsh, "A survey of constraint based scheduling systems using an artificial intelligence approach," *Artificial Intelligence in Engineering*, vol. 6, no. 2, pp. 58–73, 1991.

[57] I. Shmulevich, E. R. Dougherty, and W. Zhang, "From Boolean to probabilistic Boolean networks as models of genetic regulatory networks," *Proceedings of the IEEE*, vol. 90, no. 11, pp. 1778–1792, 2002.

[58] A. Datta, A. Choudhary, M. L. Bittner, and E. R. Dougherty, "External control in Markovian genetic regulatory networks," *Machine learning*, vol. 52, no. 1-2, pp. 169–191, 2003.

[59] M. Imani and U. M. Braga-Neto, "Point-based methodology to monitor and control gene regulatory networks via noisy measurements," *IEEE Transactions on Control Systems Technology*, vol. 27, pp. 1023 – 1035, 2019.

[60] Z. Li, P. Li, A. Krishnan, and J. Liu, "Large-scale dynamic gene regulatory network inference combining differential equation models with local dynamic Bayesian network analysis," *Bioinformatics*, vol. 27, no. 19, pp. 2686–2691, 2011.

[61] L. Qian, H. Wang, and E. R. Dougherty, "Inference of noisy nonlinear differential equation models for gene regulatory networks using genetic programming and Kalman filtering," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3327–3339, 2008.

[62] P. Spirtes, C. Glymour, R. Scheines, S. Kauffman, V. Aimale, and F. Wimberly, "Constructing Bayesian network models of gene expression networks from microarray data," 2000.

[63] L. E. Chai, S. K. Loh, S. T. Low, M. S. Mohamad, S. Deris, and Z. Zakaria, "A review on the computational approaches for gene regulatory network construction," *Computers in biology and medicine*, vol. 48, pp. 55–65, 2014.

[64] P. Li, C. Zhang, E. J. Perkins, P. Gong, and Y. Deng, "Comparison of probabilistic boolean network and dynamic Bayesian network approaches for inferring gene regulatory networks," in *BMC bioinformatics*, vol. 8, p. S13, Springer, 2007.

[65] F. Liu, S.-W. Zhang, W.-F. Guo, Z.-G. Wei, and L. Chen, "Inference of gene regulatory network based on local Bayesian networks," *PLoS computational biology*, vol. 12, no. 8, 2016.

[66] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of theoretical biology*, vol. 22, no. 3, pp. 437–467, 1969.

[67] S. Kauffman, C. Peterson, B. Samuelsson, and C. Troein, "Random Boolean network models and the yeast transcriptional network," *Proceedings of the National Academy of Sciences*, vol. 100, no. 25, pp. 14796–14799, 2003.

[68] A. Fauré, A. Naldi, C. Chaouiya, and D. Thieffry, "Dynamical analysis of a generic Boolean model for the control of the mammalian cell cycle," *Bioinformatics*, vol. 22, no. 14, pp. e124–e131, 2006.

[69] M. Imani and U. M. Braga-Neto, "Maximum-likelihood adaptive filter for partially observed Boolean dynamical systems," *IEEE Transactions on Signal Processing*, vol. 65, no. 2, pp. 359–371, 2017.

[70] K.-Y. Lau, S. Ganguli, and C. Tang, "Function constrains network architecture and dynamics: A case study on the yeast cell cycle Boolean network," *Physical Review E*, vol. 75, no. 5, p. 051907, 2007.

[71] S. Bornholdt, "Boolean network models of cellular regulation: prospects and limitations," *Journal of the Royal Society Interface*, vol. 5, no. suppl_1, pp. S85–S94, 2008.

[72] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, "The yeast cell-cycle network is robustly designed," *Proceedings of the National Academy of Sciences*, vol. 101, no. 14, pp. 4781–4786, 2004.

[73] Y. Wu, X. Zhang, J. Yu, and Q. Ouyang, "Identification of a topological characteristic responsible for the biological robustness of regulatory networks," *PLoS computational biology*, vol. 5, no. 7, 2009.

[74] Y. Xiao and E. R. Dougherty, "The impact of function perturbations in Boolean networks," *Bioinformatics*, vol. 23, no. 10, pp. 1265–1273, 2007.

[75] M. Imani and S. F. Ghoreishi, "Optimal finite-horizon perturbation policy for inference of gene regulatory networks," *IEEE Intelligent Systems*, 2020.

[76] M. Imani and U. M. Braga-Neto, "Finite-horizon LQR controller for partially-observed Boolean dynamical systems," *Automatica*, vol. 95, pp. 172–179, 2018.

[77] R. Pal, A. Datta, and E. R. Dougherty, "Optimal infinite-horizon control for probabilistic Boolean networks," *Signal Processing, IEEE Transactions on*, vol. 54, no. 6, pp. 2375–2387, 2006.

[78] M. Imani and U. Braga-Neto, "Control of gene regulatory networks using Bayesian inverse reinforcement learning," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 16, no. 4, pp. 1250–1261, 2019.

[79] H. Raiffa and R. Schlaifer, "Applied statistical decision theory," 1961.

Mahdi Imani is an Assistant Professor in the Department of Electrical and Computer Engineering at George Washington University, Washington, DC, USA. He received his Ph.D. degree in Electrical and Computer Engineering from Texas A&M University, College Station, TX, in 2019. His research interests include Machine Learning, Bayesian Statistics, and Decision Theory, with a wide range of applications from computational biology to cyber-physical systems. He is the recipient of the Association of Former Students Distinguished Graduate Student Award for Excellence in Research-Doctoral in 2019, the Best Ph.D. Student Award in ECE department at Texas A&M University in 2015, and a single finalist nominee of ECE department for the Outstanding Graduate Student Award in the college of engineering at Texas A&M University in 2018. He is also a recipient of the best paper finalist award from the 49th Asilomar Conference on Signals, Systems, and Computers, 2015.

Seyede Fatemeh Ghoreishi is a postdoctoral research fellow at the Institute for Systems Research (ISR) at the University of Maryland. She received her Ph.D. and M.Sc. degrees both in Mechanical Engineering from Texas A&M University in 2019 and 2016, respectively. She holds a minor in Applied Statistics from the Department of Statistics at Texas A&M University. She also received an M.Sc. degree in Biomedical Engineering from Iran University of Science and Technology in 2014 and a B.Sc. degree in Mechanical Engineering from the University of Tehran in 2012. She was selected as Rising Stars in Mechanical Engineering at UC Berkeley in 2020 and in Computational and Data Sciences at the Oden Institute for Computational Engineering and Sciences at the University of Texas at Austin in 2019.