Bayesian Active Learning for Drug Combinations

Mijung Park*, Marcel Nassar, *Student Member, IEEE*, and Haris Vikalo, *Member, IEEE*

***Abstract*—The dynamics of complex diseases are governed by intricate interactions of myriad factors. Drug combinations, formed by mixing several single-drug treatments at various doses, can enhance the effectiveness of the therapy by targeting multiple contributing factors. The main challenge in designing drug combinations is the highly nonlinear interaction of the constituent drugs. Prior work focused on guided space-exploratory heuristics that require discretization of drug doses. While being more efficient than random sampling, these methods are impractical if the drug space is high dimensional or if the drug sensitivity is unknown. Furthermore, the effectiveness of the obtained combinations may decrease if the resolution of the discretization grid is not sufficiently fine. In this paper, we model the biological system response to a continuous combination of drug doses by a Gaussian process (GP). We perform closed-loop experiments that rely on the expected improvement criterion to efficiently guide the exploration process toward drug combinations with the optimal response. When computing the criterion, we marginalize out the GP hyperparameters in a fully Bayesian manner using a particle filter. Finally, we employ a hybrid Monte Carlo algorithm to rapidly explore the high-dimensional continuous search space. We demonstrate the effectiveness of our approach on a fully factorial Drosophila dataset, an antiviral drug dataset for Herpes simplex virus type 1, and simulated human Apoptosis networks. The results show that our approach significantly reduces the number of required trials compared to existing methods.**

***Index Terms*—Drug combinations, expected improvement, Gaussian processes (GPs), hybrid Monte Carlo (HMC), particle filter.**

I. Introduction

FINDING effective therapeutic interventions is of vital importance for treating complex diseases including cancer, hypertension, and diabetes. Such diseases originate from biological dysfunctions in complex biological networks that are inherently robust to external perturbations. Therefore, single-drug treatments that intervene on a single target may not succeed in controlling the networks underlying the disease, and intervening on multiple targets is often required for more effective therapies [1], [2]. Furthermore, the emergence of multidrug resistant pathogens and the development of personalized medicine necessitate procedures for selecting multiple effective treatments from a big pool of available compounds and jointly optimizing their dosage levels. *Drug combinations*, mixtures that consist of multiple drugs at various dosage levels, have been effectively used in the treatment of human immunodeficiency virus [3], Herpes simplex virus [4], and various antimicrobial [5], [6], and cancer treatments [2].

A. Motivation

Traditionally, designing drug combinations relies on exhaustive empirical and clinical search which can be time consuming and costly. Consequently, there is a dire need for cost effective, rapid, and automated exploration of drug combinations. However, this problem poses the following challenges: 1) nonlinear interactions between constituent drugs result in hard-to-predict responses; and 2) the huge number of possible drug combinations leads to computationally challenging optimization tasks.

B. Prior Work

Different approaches to designing drug combinations are summarized in [7] and can be categorized into three main categories: 1) biological model-based methods; 2) biological model-free search algorithms; and 3) statistical methods. In biological model-based methods, an explicit model of the biological system is simulated to predict and optimize drug responses. For example, the authors in [8] developed a model of the Apoptosis network that governs cell death and identify potential targets for drug combinations that would elicit a desired response. However, such biological models are often unavailable or incomplete for specific biological systems and their development requires significant research effort and experimental campaigns. Biological model-free and statistical methods, on the other hand, treat the biological system as a black box and perform experiments to learn and/or optimize the system response. These methods make it possible to automate the drug combinations design. Both of these approaches are based on the notion of a *response landscape*. The response landscape is the biological system's response as a function of the drug doses used in the drug cocktail [7]. The key difference between the two methods is that statistical model-based techniques attempt to approximate the control landscape using training data and then optimize the approximated response, while the model-free methods typically optimize the control landscape on a discretized grid without explicitly constructing a model. We now review the prior work in detail.

1) Selective Tree Search Algorithms: They belong to the class of deterministic biological model-free methods. They are heuristics that avoid exhaustive search by selecting "good" candidate tree paths given the computational resource constraints. While being computationally efficient, selective tree search algorithms offer no guarantees regarding the global optimality of the constructed solution. They have been used extensively in

Manuscript received February 19, 2013; revised May 22, 2013; accepted June 18, 2013. Date of publication July 4, 2013; date of current version October 16, 2013. *Asterisk indicates corresponding author.*

*M. Park is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: mjpark@mail.utexas.edu).

M. Nassar and H. Vikalo are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: mnassar@utexas.edu; hvikalo@ece.utexas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBME.2013.2272322

data compression and communication systems where the underlying code has a tree structure [9]. In the context of drug combinations, Calzolari *et al.* [1] proposed a variant of the stack sequential algorithm (SSA) to explore the response landscape on a discretized drug grid which forms a tree with each node being a combination of the constituent drugs (with duplicate entries). The proposed variant, the stack sequential top-down (SS-TD) algorithm, combines the metric-first nature of SSA with a depth-first search at the deeper levels of the tree, with the rationale that larger combinations of drug doses are generally more effective than smaller combinations. While effective for smaller sized problems, its computational and memory requirements increase significantly with the finer discretization of the drug space and/or inclusion of large number of candidate drugs (computational and memory requirement analysis can be found in [9]). This severely limits the number of branches the algorithm can explore and reduces the effectiveness of the discovered combinations. Furthermore, this approach does not take into account the unavoidable measurement error; if significant, such errors will cripple this algorithm by propagating through all decision stages.

2) Stochastic Search Algorithms: They are biological model-free methods that explore the response landscape using a guided random walk on a discrete grid of drug doses. As an example, a Gur-game based algorithm was used for building a closed-loop feedback control system that inhibits viral infection of fibroblasts [10]. The Gur-game algorithm randomly and independently updates each drug component based on a reward mechanism that depends on the drug response of the current drug combination. The main drawback of this algorithm is that it requires normalizing the control landscape to values between 0 and 1 which might be difficult in practical situations where the response range is unknown and requires additional experiments to be determined. Recently, the author in [11] proposed a modified version of the Gur-game algorithm that overcomes aforementioned difficulties but updates only a single drug at each iteration which slows down the landscape exploration. A closely related approach to the Gur-game is based on genetic algorithms. In [2], the authors proposed a genetic algorithm variant to target cancer cells. For each generation, the algorithm selects the fittest combination and designates its nearest neighbors (i.e., a single change in the dosage of a drug) as the next generation. In this case, fitness is determined by the system response. However, unlike the Gur-game algorithm, this approach evaluates all possible neighbors and not just the ones that will likely increase the fitness. As a result, the genetic algorithm is typically less efficient than the Gur-game algorithm. Unlike the tree search algorithms, stochastic search is more robust to experimental noise; however, it still presents a significant practical challenge: how does the noise affect the algorithms' performance? This complicates the choice of the search termination criteria (it is not sufficient to keep the combination with the highest observed response since that response is noisy).

3) Statistical Methods: They construct a probabilistic model of the response landscape on a continuous drug input space. This is in contrast to the aforementioned methods that guide the landscape exploration through a discrete grid of drug dosage without explicitly modeling the response. Response surface regression (RSR), used to design antimicrobials in [5], fits the landscape to a second-order polynomial. While the RSR method requires only few data points to train, the second-order polynomial cannot model the complexity of an unknown multipeak landscape. A more advanced modeling was achieved by using neural networks in [12]. However, neural networks require large amounts of training data which significantly increases the number of required experiments. Recently, the authors proposed the use of *Gaussian Processes* (GPs) for modeling biological response landscapes [13]. GP regression is a nonparametric kernel technique that treats the regression problem in the function space. That is, a GP defines distributions over functions instead of weights as the RSR does [14]. The GP framework also yields the posterior covariance that provides the level of uncertainty about the response function estimate. The authors in [13] utilized the posterior uncertainty to select candidate drug combinations by maximizing the information gain about the response surface, which is often referred to as *info-max*. The info-max criterion, however, can be highly inefficient in a high-dimensional dose space when the goal is to find the optimum (optima) of a given biological system as opposed to estimating the entire response surface.

C. Contribution

While model-free approaches provide significant reductions in the number of required trials over random sampling, their drug space is discrete. This poses significant difficulties if the number of drugs being mixed and used for treatment is large. In such scenarios, the considered drug grid might not be fine enough, and effective drug combinations might be excluded from the search. On the other hand, while statistical methods do not suffer from this limitation, they depend on batch data such as those collected from factorial designs and thereby lack the adaptability and efficiency in data gathering procedure. In this paper, we propose a novel statistical continuous-dose solution to the drug cocktail problem based on the Bayesian sequential active learning paradigm. Under this framework, we use a GP as a surrogate model for the biological system's response. The GP models the drug response function for a target network in a flexible and nonparametric way. The continuous GP model allows us to avoid the combinatorial optimization associated with discretized drug doses and to exploit the smoothness of the biological system response. Using this probabilistic model, we then propose an adaptive learning algorithm that selects the next experiments to perform in a sequential manner so as to maximize the information gained about the optimal biological response according to the *maximum expected improvement* (MEI) criterion [15]. This criterion has been shown to be very effective in many Bayesian optimization problems [16], [17]. We address two challenges in applying the GP-MEI sequential active learning framework to drug cocktail design: 1) we account for the uncertainty about the GP hyperparameters into the MEI computation by marginalizing over them using a particle filter; and 2) we utilize the hybrid Monte Carlo (HMC) method, also referred to as Hamiltonian Monte Carlo, to explore the high-dimensional input space efficiently and thus optimize the objective function (MEI) rapidly. The model-averaging effect

provided by the proposed fully Bayesian approach allows us to deal with measurement noise by estimating its variance and providing confidence intervals on the optimized response.

II. Methods

A. Response Modeling Using a GP

We begin by introducing the observation model for drug combination responses. Let y_i be an observed biological system response at the ith trial given the input drug combination $\mathbf{x}_i$ (a vector of drug dosages). In order to account for random experimental errors, we assume the response to be a sum of a function value at $\mathbf{x}_i$ denoted by $f(\mathbf{x}_i)$ and additive Gaussian noise denoted by ϵ_i:

$$y_i = f(\mathbf{x}_i) + \epsilon_i \tag{1}$$

where $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$[1], σ^2 is the variance of the experimental error, $\mathbf{x}_i \in \mathcal{R}^d$, and d is the number of drugs. Under this model, the likelihood of an observed dataset $\mathcal{D} = \{X, \mathbf{y}\}$, where $\mathbf{y} = (y_1, \ldots, y_n) \in \mathcal{R}^n$, $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^T \in \mathcal{R}^{n \times d}$, is described by

$$p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mathbf{y}, \sigma^2 I) \tag{2}$$

where $\mathbf{f} = f(X)$ and $(\epsilon_1, \epsilon_2, \ldots, \epsilon_n)$ are assumed to be independent identically distributed.

Then, we impose a GP prior on the response function f, which defines a probability distribution over the infinite-dimensional space of functions. The GP specifies a joint Gaussian distribution over f over any finite collection of points by its mean and covariance functions [14],

$$f(\mathbf{x}_i) \sim \mathcal{N}(\mu_f(\mathbf{x}_i), k(\mathbf{x}_i, \mathbf{x}_j)) \tag{3}$$

where the mean and covariance functions are defined as

$$\begin{aligned} \mu_f(\mathbf{x}_i) &= \mathbb{E}[f(\mathbf{x}_i)], \text{ and} \\ k(\mathbf{x}_i, \mathbf{x}_j) &= \mathbb{E}[(f(\mathbf{x}_i) - \mu_f(\mathbf{x}_i))(f(\mathbf{x}_j) - \mu_f(\mathbf{x}_j))] \end{aligned} \tag{4}$$

where the covariance is defined between $\mathbf{x}_i$ and any other point $\mathbf{x}_j$. Without any prior knowledge, the mean of f is often assumed to be zero. The GP prior over f with hyperparameters θ evaluated at points in X is

$$p(\mathbf{f}|\theta) = \mathcal{N}(\mathbf{f}|0, K_\theta) \tag{5}$$

where the covariance K_θ is a matrix whose (i, j) entry is $[K_\theta]_{i,j} = k_\theta(\mathbf{x}_i, \mathbf{x}_j)$, $k_\theta(\cdot, \cdot)$ is a kernel function, and $\mathbf{f}$ is a vector of function values $((f(\mathbf{x}_1), \ldots, f(\mathbf{x}_t))^T$. Since biological responses are considered to be smooth [7], [18], [19], we use a Gaussian kernel of the form

$$k_\theta(\mathbf{x}_i, \mathbf{x}_j) = \rho \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/(2\tau)\right) \tag{6}$$

where the hyperparameters ρ and τ control the marginal variance and smoothness, respectively [14]. The GP prior, therefore, is controlled by a total of two hyperparameters, $\theta = \{\rho, \tau\}$. We drop the θ-subscript from the kernel when it is clear from context.

[1] $\mathcal{N}(\mu, \gamma)$ is the Gaussian pdf with mean μ and variance γ.

The GP prior with the exponential kernel in (6) defines an infinitely mean-square differentiable process [14]. This makes it appropriate for modeling drug response landscapes that are considered both smooth and highly nonlinear [7], [18], [19]. Furthermore, the model is fully determined by the second-order correlations between the drug responses that are parametrized by the kernel in (6). Recently, under the simplified settings of fixed drug dosages, it was shown that the response to a drug mixture of these fixed dosages can be accurately predicted using an Ising model constrained to fit the second-order moments of pairwise drug interaction [20]. The fact that GP captures all the aforementioned moments provides some evidence that it is a rich enough statistical model to capture the nonlinearity of the drug response landscape.

Note that the main goal of this paper is to personalize a drug combination based on individual patient's response to various doses. Nevertheless, if prior knowledge about the response surface is available (e.g., data collected from a cohort of patients), this knowledge can be incorporated into the proposed framework by appropriately specifying the prior mean of the GP given in (4).

B. Posterior Inference

Given the likelihood (2) and the prior (5), we compute the posterior over f at the observed points X by simply multiplying the two Gaussians

$$P(\mathbf{f}|\mathbf{y}, \vartheta) = \mathcal{N}(K\left(K + \sigma^2 I\right)^{-1}\mathbf{y}, K - K\left(K + \sigma^2 I\right)^{-1}K),$$

where $\vartheta = \{\theta, \sigma^2\}$. Furthermore, we can also obtain the posterior distribution over f at any given point $\tilde{\mathbf{x}}$ in closed form

$$P(f|\tilde{\mathbf{x}}, \mathbf{y}, \vartheta) = \int P(f|\tilde{\mathbf{x}}, \mathbf{f}, \theta)P(\mathbf{f}|\mathbf{y}, \vartheta)\mathrm{d}\mathbf{f} \tag{7}$$

$$= \mathcal{N}(\mu_\vartheta(\tilde{\mathbf{x}}), \sigma^2_\vartheta(\tilde{\mathbf{x}})) \tag{8}$$

where

$$\begin{aligned} \mu_\vartheta(\tilde{\mathbf{x}}) &= \mathbf{k}(\tilde{\mathbf{x}}, X)\left(K + \sigma^2 I\right)^{-1}\mathbf{y} \\ \sigma^2_\vartheta(\tilde{\mathbf{x}}) &= k(\tilde{\mathbf{x}}, \tilde{\mathbf{x}}) - \mathbf{k}(\tilde{\mathbf{x}}, X)\left(K + \sigma^2 I\right)^{-1}\mathbf{k}(X, \tilde{\mathbf{x}}) \end{aligned} \tag{9}$$

where $\mathbf{k}(\tilde{\mathbf{x}}, X)$ is a row vector whose ith element is $k(\tilde{\mathbf{x}}, \mathbf{x}_i)$, and $\mathbf{k}(X, \tilde{\mathbf{x}}) = \mathbf{k}(\tilde{\mathbf{x}}, X)^T$.

C. Hyperparameter Marginalization Using Particle Filtering

The posterior over f in (9) depends on the hyperparameters ϑ. If the probability distribution of the hyperparameters has a sharp peak at its most likely value, i.e., $P(\vartheta|\mathbf{y}) \approx \delta(\vartheta - \vartheta_{ml})$, we obtain the marginal posterior over f as

$$\begin{aligned} p(f|\tilde{\mathbf{x}}, \mathbf{y}) &= \int p(f|\tilde{\mathbf{x}}, \mathbf{y}, \vartheta)p(\vartheta|\mathbf{y})\mathrm{d}\vartheta \\ &\approx p(f|\tilde{\mathbf{x}}, \mathbf{y}, \vartheta_{ml}). \end{aligned} \tag{10}$$

A widely used method to set ϑ is by maximizing the marginal likelihood of ϑ (or the so-called *evidence*)

$$\mathcal{E}(\vartheta) = p(\mathbf{y}|\vartheta) = \mathcal{N}(0, K_\theta + \sigma^2 I) \tag{11}$$

where maximizing $p(\mathbf{y}|\vartheta)$ is consistent with maximizing $p(\vartheta|\mathbf{y})$ under a broad uniform prior on ϑ via Bayes' rule.

Although maximizing the evidence is computationally appealing, in practice, we will not have such a peaky posterior on ϑ unless we have large amounts of data to support it. Point estimates obtained by maximizing (11) ignore the uncertainty in the hyperparameters, which might be significant in the scarce data regime considered here. In addition, the optimization is not convex and is likely to have multiple local maxima. To circumvent this problem, we adopt a fully Bayesian inference and marginalize out the hyperparameters using Monte Carlo integration

$$p(f|\tilde{\mathbf{x}},\mathbf{y}) = \int p(f|\tilde{\mathbf{x}},\mathbf{y},\vartheta)p(\vartheta|\mathbf{y})\mathrm{d}\vartheta \tag{12}$$

$$\approx \sum_i p(f|\tilde{\mathbf{x}},\mathbf{y},\vartheta_i)\mathcal{E}(\vartheta_i). \tag{13}$$

Note that the marginal posterior over f is a mixture of GPs.

We first draw samples of ϑ from the evidence and then sum up all the conditional posteriors to obtain the marginal distribution. However, we need to do this in every trial, which is computationally expensive. Recent work developed an iterative algorithm using Bayesian Monte Carlo under a GP model to marginalize hyperparameters [21]. However, the approach uses a fixed sample set of hyperparameters and only updates the weights of the samples, which could suffer from the well-known progressive degeneracy in samples. Here, we adopt a *resample-move particle filter* proposed in [22], which effectively overcomes the degeneracy in particles by adding a Markov Chain Monte Carlo (MCMC) move at the last stage. The details are given below.

Let $\mathcal{D}_t$ denote the data collected up to time *t*, i.e., $\mathcal{D}_t = \{\mathbf{x}_i, y_i\}_{i=0}^t$. Suppose the evidence $p(\vartheta_t|\mathcal{D}_t)$ is available at time t. We compute the marginal posterior mean and covariance at time *t* from (12)

$$\mu_t(\tilde{\mathbf{x}}) = \sum_{\vartheta_t} p(\vartheta_t|\mathcal{D}_t)\mu_{\vartheta_t}(\tilde{\mathbf{x}})$$

$$\sigma_t^2(\tilde{\mathbf{x}}) = \sum_{\vartheta_t} p(\vartheta_t|\mathcal{D}_t)\left(\sigma_{\vartheta_t}^2(\tilde{\mathbf{x}}) + \mu_{\vartheta_t}^2(\tilde{\mathbf{x}})\right) - \mu_t^2(\tilde{\mathbf{x}}) \tag{14}$$

where the latter equation is based on the law of total variance (variance decomposition formula).

1) Prediction: We first evaluate the prediction step as

$$p(\vartheta_{t+1}|\mathcal{D}_t) = \int p(\vartheta_t|\mathcal{D}_t)p(\vartheta_{t+1}|\vartheta_t)d\vartheta_t.$$

However, since there is no specific transition model in our setup, we assume $p(\vartheta_{t+1}|\mathcal{D}_t) \simeq p(\vartheta_t|\mathcal{D}_t)$.

2) Resampling: Using a new input/output pair $\{\mathbf{x}, y\}$, we update the evidence according to

$$p(\vartheta_{t+1}|\mathcal{D}_t,\{\mathbf{x},y\}) \propto p(y|\vartheta_{t+1},\mathcal{D}_t,\mathbf{x})p(\vartheta_{t+1}|\mathcal{D}_t). \tag{15}$$

The first term on the right is the importance weight

$$p(y|\vartheta_{t+1},\mathcal{D}_t,\mathbf{x}) \simeq p(y|\vartheta_t,\mathcal{D}_t,\mathbf{x}),$$
$$= \int p(f|\mathbf{x},\mathcal{D}_t,\vartheta_t)p(y|f,\mathbf{x})df.$$

Since the integrand is a product of two Gaussians, $f|\mathbf{x},\mathcal{D}_t,\vartheta_t \sim \mathcal{N}(\mu_{\vartheta_t}(\mathbf{x}),\sigma_{\vartheta_t}^2(\mathbf{x}))$ and $y|f,\mathbf{x} \sim \mathcal{N}(f(\mathbf{x}),\sigma_t^2)$, the importance weight is simply given by

$$p(y|\vartheta_{t+1},\mathcal{D}_t,\mathbf{x}) \simeq \mathcal{N}(\mu_{\vartheta_t}(\mathbf{x}),\sigma_{\vartheta_t}^2(\mathbf{x}) + \sigma_t^2). \tag{16}$$

Based on (16), we resample the particles.

3) MCMC Sampling: We use the Metropolis Hastings (MH) algorithm for MCMC sampling. To carry out the MH sampling, we first sample from a multivariate Gaussian centered on the current particles ϑ_{cp} of the Markov chain, $\vartheta^* \sim \mathcal{N}(\vartheta_{cp},\Gamma)$, where Γ is the diagonal matrix whose diagonal entries are the variance of particles in the previous trial. We assume a noninformative hyperprior $p(\vartheta)$, taken to be uniform over a fairly broad range of values. Then, we compute $\alpha = \frac{q(\vartheta*)}{q(\vartheta_t)}$, with $q(\vartheta) = p(\vartheta|\mathcal{D}_t)$. With probability $\min(1,\alpha)$, we accept the proposal: $\vartheta_{t+1} = \vartheta^*$; otherwise, we set $\vartheta_{t+1} = \vartheta_{cp}$.

4) Update Posterior Moments: Once we have the new set of particles for hyperparameters at time $t+1$, we update the posterior mean and variance according to (14).

D. Expected Improvement for Drug Combinations Selection

Having defined a statistical model and developed the Bayesian framework for modeling the response landscape, we describe our closed-loop procedure for efficient selection of experiments. In particular, we employ an active learning framework typically referred to as Bayesian optimization. The algorithm selects experiments which maximize a popular criterion in Bayesian optimization literature, MEI [15]. The MEI criterion selects the input drug combination candidate that yields the most information about the optimal value of the response surface given the collected data $\mathcal{D}_t$. The expected improvement (EI) is given by,

$$\begin{aligned}\mathrm{EI}(\tilde{\mathbf{x}}) &= \mathbb{E}_{f_{\tilde{\mathbf{x}}}|\mathcal{D}_t}\left[\max(f(\tilde{\mathbf{x}}) - f^*, 0)\right] \\ &= \int_{f^*}^{\infty}(f(\tilde{\mathbf{x}}) - f^*)\mathcal{N}(f(\tilde{\mathbf{x}})|\mu_{\tilde{\mathbf{x}}},\sigma_{\tilde{\mathbf{x}}}^2)\mathrm{d}f(\tilde{\mathbf{x}}) \\ &= (\mu(\tilde{\mathbf{x}}) - f^*)\Phi\left(\frac{\mu(\tilde{\mathbf{x}}) - f^*}{\sigma(\tilde{\mathbf{x}})}\right) + \sigma(\tilde{\mathbf{x}})\phi\left(\frac{\mu(\tilde{\mathbf{x}}) - f^*}{\sigma(\tilde{\mathbf{x}})}\right)\end{aligned} \tag{17}$$

where Φ is the cumulative density function of the standard normal distribution, ϕ is the probability density function (pdf) of the standard normal distribution, and μ and σ are the posterior mean and the standard deviation at any point $\tilde{\mathbf{x}}$ in (9) (we dropped ϑ for notational simplicity). Moreover, f^* denotes the optimal value of the current posterior mean at observed points (7). Finally, we select the next input maximizing the EI as

$$\mathbf{x}^* = \arg\max_{\tilde{\mathbf{x}}\in\mathcal{X}} \mathrm{EI}(\tilde{\mathbf{x}}). \tag{18}$$

The candidate set $\mathcal{X}$ determines possible values of drug combinations $\mathbf{x}$ that we can choose from. If the candidates can take unrestricted values, then the set $\mathcal{X}$ becomes $\mathcal{R}^n$. On the other hand, if the drug dosage levels are restricted to a grid of predetermined drug dosage levels as in the model-free methods, then the set $\mathcal{X}$ is a finite set of drug combinations. It should be stressed that in the latter case, unlike the model-free search methods, the search on the finite $\mathcal{X}$ is purely computational and does not involve any biological experiments. Furthermore, it makes use

of the smoothness conditions on the response landscape present in biological systems and imposed by the GP model.

As seen from (18), the next drug combination to test is the solution to a "computational" optimization problem involving the maximization of the EI criterion. The EI objective presents challenges that might limit the effectiveness of standard numerical optimization algorithms: 1) in general, EI is not convex; and 2) EI might be high dimensional when a large number of potential drugs is being tested. As a result, we propose an alternate optimization approach based on HMC sampling.

E. Optimization of EI Using HMC

From (17), it can be seen that $\mathrm{EI}(\tilde{\mathbf{x}}) \geq 0$. As a result, it can be treated as a pdf (up to a normalization constant) and sampled using Monte Carlo techniques. Given that EI is peaky, as is expected when dealing with highly nonlinear response surfaces, the generated samples will be close to the EI maximizer (the mode of the sampling pdf). If we let $\mathcal{S} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{M_S}\}$ be the set of M_S samples generated using the HMC, then the optimizer (the next drug combination to test) is given by

$$\mathbf{x}^* = \arg\max_{\tilde{\mathbf{x}} \in \mathcal{S}} \mathrm{EI}(\tilde{\mathbf{x}}). \tag{19}$$

Note that EI is evaluated at each $\mathbf{x} \in \mathcal{S}$ during the HMC sampling and does not require any additional computations.

HMC is a Markov Chain Monte Carlo method which relies on Hamiltonian dynamics to generate distant proposals for the Metropolis algorithm, thus avoiding slow exploration of the input space resulting from random-walk proposals. As a results, the HMC sampler efficiently draws samples from high-dimensional target distribution. We now briefly summarize HMC sampling.

Having origins in physics, HMC integrates MCMC simulations of the distribution of system states with a deterministic description which represents total energy of the system as the sum of *potential* energy and *kinetic* energy. Suppose there is a frictionless puck sliding over a surface of target distribution. The Hamiltonian of the system is the sum of potential energy of the puck (determined by position of the puck) and the kinetic energy of the puck (determined by momentum of the puck). In non-physical MCMC applications, the position corresponds to variables of interest—in our case, the drug combinations. The potential energy is defined as minus log of target distribution

$$U(\mathbf{q}) = -\log p(\mathbf{q}), \quad \mathbf{q} : \text{position}$$

where our target distribution from (17) is

$$p(\mathbf{q}) = (\mu(\mathbf{q}) - f^*)\Phi\left(\frac{\mu(\mathbf{q}) - f^*}{\sigma(\mathbf{q})}\right) + \sigma(\mathbf{q})\phi\left(\frac{\mu(\mathbf{q}) - f^*}{\sigma(\mathbf{q})}\right).$$

In addition to the position variables, we introduce auxiliary momentum variables which are typically assumed to be multivariate (zero-mean) Gaussian. The corresponding kinetic energy (minus log of the zero-mean multivariate Gaussian) is defined as

$$K(\mathbf{p}) = \frac{\mathbf{p}^T M^{-1} \mathbf{p}}{2}, \quad \mathbf{p}\text{: momentum}$$

where M is a symmetric positive definite mass matrix (to be set). Consequently, the Hamiltonian can be written as

$$H(\mathbf{q}, \mathbf{p}) = U(\mathbf{q}) + K(\mathbf{p}).$$

The derivative of the Hamiltonian with respect to the fictitious time t defines the dynamics

$$\begin{aligned} \frac{\partial q_i}{\partial t} &= \frac{\partial H}{\partial p_i} = [M^{-1}p]_i \\ \frac{\partial p_i}{\partial t} &= -\frac{\partial H}{\partial q_i} = -\frac{\partial U}{\partial q_i} \end{aligned} \tag{20}$$

where it is straightforward to compute

$$-\frac{\partial U}{\partial q_i} = e^{U(q_i)}\left[\mu'(q_i)\Phi\left(\frac{\mu(q_i) - f^*}{\sigma(q_i)}\right) + \sigma'(q_i)\phi\left(\frac{\mu(q_i) - f^*}{\sigma(q_i)}\right)\right],$$

where $\mu'(q_i) = \frac{\partial}{\partial q_i}\mu(q_i)$ and $\sigma'(q_i) = \frac{\partial}{\partial q_i}\sigma(q_i)$.

In computer simulations, we can compute the Hamiltonian dynamics only approximately by discretizing time using a small step size (τ). One of the widely used methods is the *leapfrog* method, which iterates the following steps to walk around some contours of the target distribution [23]:

$$\begin{aligned} p_i\left(t + \frac{\tau}{2}\right) &= p_i(t) - \frac{\tau}{2}\frac{\partial U(q(t))}{\partial q_i} \\ q_i(t + \tau) &= q_i(t) + \tau\frac{p_i(t + \frac{\tau}{2})}{M_{ii}} \\ p_i(t + \tau) &= p_i\left(t + \frac{\tau}{2}\right) - \frac{\tau}{2}\frac{\partial U(q(t + \tau))}{\partial q_i}. \end{aligned} \tag{21}$$

The iterations start with a half step for p, following a full step for q, and then proceed with another half step for p.

After L leapfrog steps, we accept/reject the proposed state in a Metropolis step using the joint probability $p(\mathbf{q}, \mathbf{p})$, and then add Gibbs moves so that we can effectively sample from isolated modes. In practice, we need to tune the HMC sampler by setting the mass matrix M in the expression for kinetic energy, the discretization step size (τ), and the leapfrog trajectory length (L) properly.

Compared to widely used Monte Carlo sampling algorithms (e.g., MH or Gibbs sampling) that are slow in exploring high-dimensional spaces, HMC enables us to rapidly search the high-dimensional dose space. Furthermore, most prior work in the Bayesian optimization literature uses gradient-based optimization methods to find the EI's optimum, which could be trapped in local optima when only a limited amount of data are available. However, the proposed HMC sampler can move from one mode to another under appropriate tuning of the parameters, making it an effective way to optimize the nonconvex EI objective.

F. Complete Algorithm

Recall that our posterior over f is a mixture of GPs, which makes it hard to draw samples from EI via the HMC framework. For example, if we use 100 particles for ϑ, then we will have

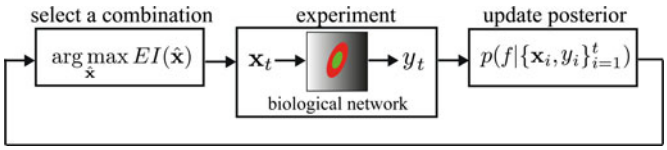

Fig. 1. Schematic of Bayesian active learning of drug combinations. The sequential close-loop experiment consists of: 1) selecting an optimal drug combination by maximizing expected improvement (left); and 2) measuring the response y_t to the selected combination $\mathbf{x}_t$ (middle); updating the posterior over the response function given all the data collected so far (right).

100 target distributions (EI with different ϑ) to draw samples from.

To address this issue, we approximate the marginal posterior by a single GP whose first two moments match those of (12) [24]. Such an approximation is justified by the fact that if the approximating distribution is Gaussian, then its Kullback–Leibler distance from the marginal posterior distribution is minimized when its first and second moments match those of the marginal posterior distribution [25]. Thus, we use the approximate posterior at each time t given by

$$p(f|\tilde{\mathbf{x}}, \mathcal{D}_t) \approx \mathcal{N}(\mu_t(\tilde{\mathbf{x}}), \sigma_t^2(\tilde{\mathbf{x}})) \tag{22}$$

and then search the optimal combination that maximizes EI under the approximation by HMC sampling. Note that the approximate posterior is not the same as fixing the hyperparameters to certain values and using a Gaussian marginal posterior as in (10). The former takes into account hyperparameter uncertainty (via particle filtering), while the latter ignores the uncertainty. The benefit of hierarchical model and fully Bayesian inference is illustrated in Section III.

The complete proposed algorithm for sequential active learning for drug combinations is summarized below.

Algorithm 1 Sequential active learning for drug combinations

Given a set of particles from $p(\vartheta_t|\mathcal{D}_t)$ at t,
repeat
1. Approximate posterior over f using current particles, given by (14) and (22).
2. Select a candidate input $\mathbf{x}$ using MEI via HMC.
3. Measure y in response to $\mathbf{x}$.
4. Resample particles for ϑ from (15) with importance weight given by (16).
5. Carry out MH sampling centered at the resampled particles.
until a stopping criterion is satisfied

In Bayesian optimization and active learning literature, the stopping criterion is typically determined by resource limitations. However, the algorithm can also be stopped when the posterior variance at the optimal input is below a certain threshold or if the change in the optimum value from one iteration to the other is below a given threshold.

III. Results

A. Drosophila Dataset (In Vivo)

We ran three different algorithms on a factorial dataset, published in [1], of scalar fitness scores, called z-scores (summarizing maximal heart rate, exercise capacity, and survival of aged Drosophila flies) in response to 81 different combinations of four drugs (with three doses for each drug including zero dose).

The goal was to find the three most effective drug combinations among the 81 options using as few experimental trials as possible.

Fig. 2(A) shows the average (computed over 100 repetitions) of the estimated z-scores as a function of the number of trials. The SS-TD algorithm considered in [1] (here shown in pink) required 27 trials to find the top three combinations (24 trials for SS). Note that since the SS-TD algorithm does not consider any measurement noise, there is no randomness in that method and therefore no need to repeat it multiple times to quantify the performance. The info-max (in black) method, proposed in [13], required 27 trials on average to achieve the same goal. By attempting to learn the whole response function rather than focusing on the optima, the info-max method is wasting experiments and hence does not improve over SS-TD. When applying the modified Gur-game algorithm [11] to this dataset, it required on average 23 trials to find the top three combinations. However, it did not always succeed. We found that its success rate[2] was typically 0.8 (in contrast, the proposed algorithm always found the top three combinations).

On the other hand, our proposed method using 100 particles (in red solid, labeled as MEIfb) required only 14 trials to identify the top three drug combinations. Note that since there are only 81 possible combinations, there was no need to perform the HMC sampling to optimize the MEI criterion; instead, we computed the criterion on the 81 input points and selected the input corresponding to the highest value. To further demonstrate the benefits of including the hyperparameter uncertainties in the MEI criterion, we also demonstrate the performance of the MEI approach that estimates the hyperparameters using maximum marginal likelihood (shown in red dotted trace, labeled as MEIeb). This technique on average required 22 trials on to achieve the same goal. Note that we used seven randomly drawn inputs for the info-max method and MEIeb to set the initial hyperparameter values, and then proceeded with the simulated experiments. Previous approaches include initial preacquisition of points in a Latin hypercube design as many points as ten times the dimensionality of the input space [16], [17]. However, since we have only 80 measurements (at the fixed input points) in the Drosophila dataset, we used randomly chosen (approximately) 10% of the total datapoints for the initial points. For this reason, the trial number starts with eight for these two methods.

B. Antiviral Drug Combinations for HSV-1 (In Vivo)

We also tested the performance of our algorithm when applied to finding the best drug combination for curing Herpes simplex virus type 1. The dataset was originally published in [4], where six drugs were considered with two different dose level for each drug. However, the dataset given in [4, Table 1] shows only 32 different fractional factorial combinations

[2]Success rate is defined as the probability of finding the top three combinations from 100 independent repetitions.

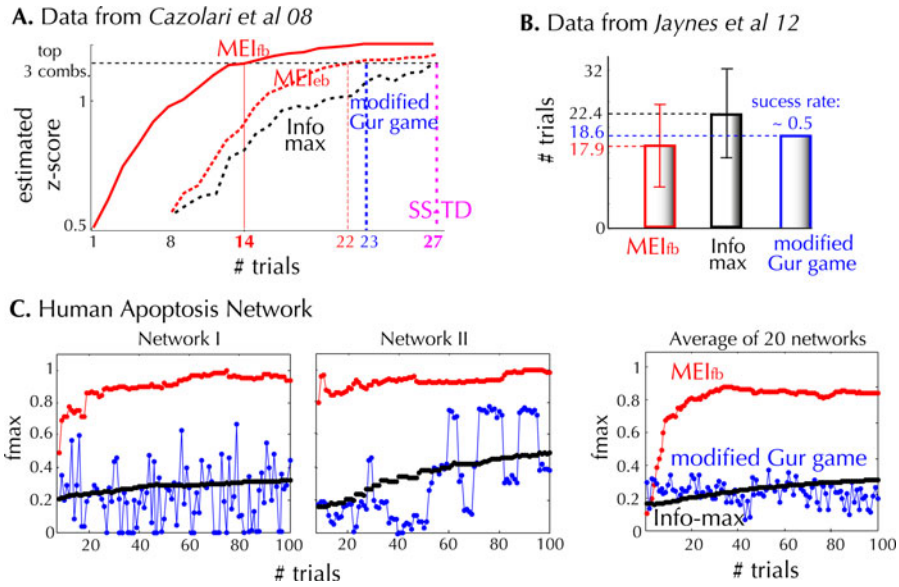

Fig. 2. (A) Performance test on the *Drosophila* fully factorial dataset. (B) Performance test on six antiviral drugs for *Herpes simplex virus* type 1 (HSV-1) dataset. (C) Simulations using six drugs on the human *Apoptosis* networks. (See the text).

(excluding zero doses) with their corresponding biological measurements. Therefore, we restrict the search space to the 32 given combinations.

Fig. 2(B) shows the average number of trials needed to find the optimal combination using the proposed method (using 100 particles), the info-max method, and the modified Gur game. The proposed method required approximately 15% less trials than the info-max method required. The modified Gur-game algorithm, on the other hand, required 18.6 trials on average to obtain the optimal combination with a low success rate of 0.51 (as before, the proposed method always found the optimal combination).

C. Simulations on the Apoptosis Network (In Silico)

Finally, we conducted a computational study of multiple interventions on the human Apoptosis network that governs the process of programmed cell death. The Apoptosis network has important applications to the design of therapies for complex diseases such as cancer and Alzheimer. We used the computational model developed in [8] based on the Apoptosis network "hsa04210" of the KEG database. This network has 11 input nodes and a single output node that determines cell's life or death. The dynamics in this network is governed by the Hill equation leading to a highly nonlinear input/output relationship [26].

With the goal of inducing cell death, we investigated the design of optimal drug interventions on (randomly selected) 6 input nodes. We compared the performance of the proposed algorithm with that of the modified Gur-game algorithm for a continuous dose space between 0 and 1. As required by the modified Gur-game algorithm, we discretized the drug dosage evenly to four levels: 0, 0.3, 0.6, and 0.9 (resulting in 4096 possible combinations). For the info-max method, we used the same number of points defined on the input space for the sake of computational tractability. The proposed algorithm used 100 particles for estimating the hyperparameters in each trial and drew 100 candidate samples from the HMC sampler with a leapfrog trajectory of length $L = 10$, step size $\tau = 0.02$, and diagonal mass matrix $M = 36I$ (these were heuristically chosen; see [23] for details on how to tune the HMC sampler). We computed the EI criterion at the 100 candidate drug combinations, and then chose the one corresponding to the highest value as the next input.

In Fig. 2(C) (left), we showed the normalized output values[3] as a function of the number of trials. The two random instantiation of the Apoptosis network are denoted as *Network I* and *Network II*. In Network I, the modified Gur-game (blue) approach produced drug combinations that were 40% less effective than those produced by the proposed method. In fact, due to requiring a discrete drug-dose space, the Gur-game's search space does not even include the particular combination that elicited the highest response. On the other hand, while not requiring a discrete drug-dose space, the info-max method (black) also failed to find the combination for the highest response. This can be attributed to its criterion targeting the reduction of the posterior variance of the entire response surface rather than focusing on the optimum (the true objective in drug combination design). In Network II, the drug combination produced by the Gur game is slightly better than that produced for Network I (although it took more experiments to achieve it). The proposed algorithm performs consistently well on both networks. To get a better idea of how this algorithms will perform in practice, Fig. 2(C) (right) shows the average normalized output responses predicted by the three methods for 20 instantiations of the Apoptosis network. While the average normalized output values of both modified

[3]The normalization is done by dividing the response due to a drug combination designed by each method by the highest response among 100 repetitions (due to the complexity of the Apoptosis network a closed form for the maximum is not known).

Gur game and info-max algorithms were below 0.5, the average output value of the proposed method was approximately 0.8 even with a small number of trials such as 50. We also tested the genetic algorithm on the Apoptosis network (results reported in [13]). The genetic algorithm (for a fixed number of generations) performs significantly worse than random sampling as well as the info-max algorithm under the presence of noise.

D. Discussion

By employing a fully Bayesian hierarchical drug response model, we reduce the drug combination design problem with exponentially many possibilities into a relatively low dimensional hyperparameter inference problem (typically $\ll 10$, in our case only 3). To make the algorithm computationally tractable to run in practical medical settings, we employ the following algorithmic modification: 1) we employ a particle filter to take into account the uncertainty in the hyperparameters; and 2) we use an HMC method to rapidly optimize the EI criterion over the high-dimensional drug-dose space, effectively avoiding getting stuck in local optima. This allows the proposed algorithm to rapidly and steadily converge to optimal combinations (as demonstrated by extensive simulations) and produce confidence intervals around the produced solutions. In comparison, as shown in the simulations, the state-of-the-art Gur-game algorithm does not produce as effective combinations and typically oscillates significantly due to the random walk it performs while exploring the drug space. These random oscillations make it difficult to judge when enough experiments have been performed or how far from the optimum are the produced combinations.

While there are no theoretical guarantees on the convergence of the proposed algorithm, its observed rapid convergence can be attributed to the approximate inference it is performing: as the number of data points grows, the posterior of the response surface near the maximum value becomes narrower and our estimate of the maximum converges to the global optimum.

IV. Conclusion

Bayesian inference is a powerful statistical framework that remains largely unexploited in designing drug combinations. Here, we develop a Bayesian active learning framework which finds optimal drug combinations by using the expected improvement criterion to guide closed-loop experiments. Our approach allows us to handle measurement noise and exploit the smoothness of the drug response surface without discretizing the drug dosage space that might introduce a systematic bias in the drug combination solution. Test results on two real datasets and on a simulated human Apoptosis network show that our approach significantly reduces the number of required drug trials compared to the prior state-of-the-art methods.

References

[1] D. Calzolari, S. Bruschi, L. Coquin, J. Schofield, J. D. Feala, J. C. Reed, A. D. McCulloch, and G. Paternostro, "Search algorithms as a framework for the optimization of drug combinations," *PLoS Comput. Biol.*, vol. 4, no. 12, p. e1000249, 2008.

[2] R. G. Zinner, B. L. Barrett, E. Popova, P. Damien, A. Y. Volgin, J. G. Gelovani, R. Lotan, H. T. Tran, C. Pisano, G. B. Mills, L. Mao, W. K. Hong, S. M. Lippman, and J. H. Miller, "Algorithmic guided screening of drug combinations of arbitrary size for activity against cancer cells," *Molecular Cancer Therapeut.*, vol. 8, no. 3, pp. 521–532, 2009.

[3] A. S. Perelson, P. Essunger, Y. Cao, M. Vesanen, A. Hurley, K. Saksela, M. Markowitz, and D. D. Ho, "Decay characteristics of HIV-1-infected compartments during combination therapy," *Nature*, vol. 387, no. 6629, pp. 188–191, May 1997.

[4] J. Jaynes, X. Ding, H. Xu, W. K. Wong, and C. Ho, "Application of fractional factorial designs to study drug combinations," *Statist. Med.*, vol. 32, pp. 307–318, 2012.

[5] C. H. Chen, V. Gau, D. D. Zhang, J. C. Liao, F. Wang, and P. K. Wong, "Statistical metamodeling for revealing synergistic antimicrobial interactions," *PLoS ONE*, vol. 5, no. 11, p. e15472, 2010.

[6] B. G. Small, B. W. McColl, R. Allmendinger, J. Pahle, G. López-Castejón, N. J. Rothwell, J. Knowles, P. Mendes, D. Brough, and D. B. Kell, "Efficient discovery of anti-inflammatory small-molecule combinations using evolutionary computing," *Nat. Chem. Biol.*, vol. 7, no. 12, pp. 902–908, Dec. 2011.

[7] J. D. Feala, J. Cortes, P. M. Duxbury, C. Piermarocchi, A. D. McCulloch, and G. Paternostro, "Systems approaches and algorithms for discovery of combinatorial therapies," *Wiley Interdisciplinary Rev. Syst. Biol. Med.*, vol. 2, no. 2, pp. 181–193, 2010.

[8] D. Calzolari, G. Paternostro, P. L. Harrington, Jr, C. Piermarocchi, and P. M. Duxbury, "Selective control of the apoptosis signaling network in heterogeneous cell populations," *PLoS ONE*, vol. 2, no. 6, p. e547, 2007.

[9] J. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Trans. Commun.*, vol. COM-32, no. 2, pp. 169–176, Feb. 1984.

[10] P. K. Wong, F. Yu, A. Shahangian, G. Cheng, R. Sun, and C. Ho, "Closed-loop control of cellular functions using combinatory drugs guided by a stochastic search algorithm," *Proc. Natl. Acad. Sci.*, vol. 105, no. 13, pp. 5105–5110, Apr. 2008.

[11] B. Yoon, "Enhanced stochastic optimization algorithm for finding effective multi-target therapeutics," *BMC Bioinf.*, vol. 12, Suppl 1, p. S18, 2011.

[12] K. Takayama, M. Fujikawa, Y. Obata, and M. Morishita, "Neural network based optimization of drug formulations," *Adv. Drug. Del. Rev.*, vol. 55, no. 9, pp. 1217–1231, Sep. 2003.

[13] M. Park, M. Nassar, B. Evans, and H. Vikalo, "Adaptive experimental design for drug combinations," in *Proc. IEEE Workshop Statist. Signal Process.*, Aug. 2012, pp. 712–715.

[14] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[15] J. Mockus, "Application of Bayesian approach to numerical methods of global and stochastic optimization," *J. Global Optim.*, vol. 4, no. 4, pp. 347–365, Jun. 1994.

[16] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, Dec. 1998.

[17] D. J. Lizotte, "Practical Bayesian optimization" Ph.D. dissertation, Dept. Comput. Sci., Univ. Alberta, Edmonton, AB, Canada, 2008.

[18] R. J. Tallarida, "Drug synergism: Its detection and applications," *J. Pharmacol. Exper. Therapeut.*, vol. 298, no. 3, pp. 865–872, 2001.

[19] C. F. Minto, T. W. Schnider, T. G. Short, K. M. Gregg, A. Gentilini, and S. L. Shafer, "Response surface model for anesthetic drug interactions," *Anesthesiology*, vol. 92, no. 6, pp. 1603–1616, 2000.

[20] K. Wood, S. Nishida, E. D Sontag, and P. Cluzel, "Mechanism-independent method for predicting response to multidrug combinations in bacteria," *Proc. Nat. Acad. Sci.*, vol. 109, no. 30, pp. 12254–12259, 2012.

[21] M. A. Osborne, S. J. Roberts, A. Rogers, S. D. Ramchurn, and N. R. Jennings, "Towards real-time information processing of sensor network data using computationally efficient multi-output Gaussian processes," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, 2008, pp. 109–120.

[22] W. R. Gilks and C. Berzuini, "Following a moving target—Monte Carlo inference for dynamic Bayesian models," *J. R. Statist. Soc. Series B (Statist. Methodol.)*, vol. 63, no. 1, pp. 127–146, 2001.

[23] R. M. Neal, *Handbook of Markov Chain Monte Carlo*. Boca Raton, FL, USA: CRC Press, 2010.

[24] M. Park and J. W. Pillow, "Bayesian active learning with localized priors for fast receptive field characterization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2357–2365.

[25] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[26] J. Weiss, "The hill equation revisited: Uses and misuses," *FASEB J.*, vol. 11, pp. 835–841, 1997.

Authors' photographs and biographies not available at the time of publication.