

Optimization of Computationally Expensive Simulations with Gaussian Processes and Parameter Uncertainty: Application to Cardiovascular Surgery

Jing Xie¹, Peter I. Frazier¹, Sethuraman Sankaran², Alison Marsden², and Saleh Elmohamed³

Abstract—In many applications of simulation optimization, the random output variable whose expectation is being optimized is a deterministic function of a low-dimensional random vector. This deterministic function is often expensive to compute, making simulation optimization difficult. Motivated by an application in the design of grafts for heart surgery with uncertainty about input parameters, we use Bayesian methods to design an algorithm that exploits this random vector’s low-dimensionality to improve performance.

I. INTRODUCTION

Motivated by an application in cardiovascular surgery with parameter uncertainty, we develop a new method for optimization of an objective function whose value is the average of the output of a computationally expensive simulator, where the input is varied across some low-dimensional space. We use Bayesian methods, in which inference based on a Gaussian process prior learns the behavior of the computationally expensive simulator across the input space and tracks our uncertainty about values at unevaluated points, and value of information calculations tell us at which inputs it would be most valuable to evaluate the simulator next.

The application that we consider is the design of idealized bypass graft models under uncertain shape design variables and unsteady flow, using a simulation of blood-flow near the graft. Our goal in using simulation is to compute the optimal graft attachment angles that minimize the area of low wall-shear stress (WSS). Previously, non-Bayesian methods (surrogate management framework SMF) [1] were used to perform robust shape optimization in cardiovascular simulations [2], [3], [4]. To account for uncertainties, a stochastic collocation method [4] was coupled with the SMF framework. This method has the ability to converge to a mesh local optima for Lipschitz continuous functions. [3] demonstrated that accounting for implementation and measurement uncertainties affects the optimal graft attachment angle.

In this work, we investigate the expected performance of the design variables under low-dimensional uncertainties. We

assume that the area of low WSS regions, which is our simulation output, depends on the attachment angles incorporating a “fudge factor” to account for random implementation error, the random stenosis radius, and the random inflow velocity. The random output variable is thus a deterministic function of a low-dimensional random vector. Evaluation of this deterministic function is expensive, and its derivative information is unavailable. Our goal is to optimize the expectation of this output variable (or its variant) by allocating simulation effort efficiently across different values of the random vector.

This type of problems arises in many applications of simulation optimization. [5] and [6] perform robust optimization of the design of biomechanical devices by employing an empirical best linear unbiased prediction of the structural response, where they also incorporate environmental variables.

In its attempt to evaluate the expectation (integral) of an implicit function, this work is closely related to the Bayesian Quadrature [7] or the Bayesian Monte Carlo method [8], which models the integrand using Gaussian process (GP) [9], and then performs inference about the integral by taking advantage of the analytical convenience of the GP models.

To design a strategy that samples efficiently, we employ a Bayesian approach, in which we begin with a GP prior distribution on the response function, updating this prior distribution based on sampling information, evaluate the expectation of the response function under uncertain model inputs and unsteady flow, and use “value of information” computations to decide how to best allocate sampling effort.

The value of information approach [10], [11], [12], [13] has been used in Bayesian ranking and selection [14], where sampling decisions are made to achieve the highest potential for improving the final selection decision. The current work incorporates this technique and proposes a one-step lookahead sampling procedure. Procedures of this type are commonly used in Bayesian experimental designs, e.g., global optimization [15], [16], [17]. They are also called knowledge-gradient policies [18].

II. PROBLEM FORMULATION

In this section we formulate the Bayesian shape optimization problem of idealized bypass graft models with unsteady flow. This problem is studied in [3], which uses the stochastic collocation technique to incorporate and study the effects of input uncertainties, and applies a derivative-free optimization method to perform robust shape design.

P.I. Frazier and J. Xie were supported by AFOSR FA9550-11-1-0083

¹J. Xie and P.I. Frazier are with the School of Operations Research and Information Engineering, Cornell University, Ithaca NY 14850, USA jx66 at cornell.edu pf98 at cornell.edu

²S. Sankaran and A. Marsden are with the Department of Mechanical and Aerospace Engineering, University of California San Diego, San Diego, CA 92121, USA sesankaran at eng.ucsd.edu abbangal at ucsd.edu amarsden at eng.ucsd.edu

³S. Elmohamed is with (Saleh, please fill in) m.a.saleh.elmohamed@gmail.com

In this problem, the design variables are the target anastomosis angles x_1 and x_2 given to the surgeon. Given these target values, the actual angles of a bypass graft constructed in a surgery are not x_1 and x_2 , but instead $\theta_1 = x_1 + \delta_1$ and $\theta_2 = x_2 + \delta_2$, where δ_1 and δ_2 are the implementation errors introduced during surgery. As shown in Figure 1, we denote by r and v the stenosis radius and the inflow velocity respectively. We then write $x = (x_1, x_2)$, $\delta = (\delta_1, \delta_2)$, $\theta = x + \delta$ and $\omega = (r, v)$.

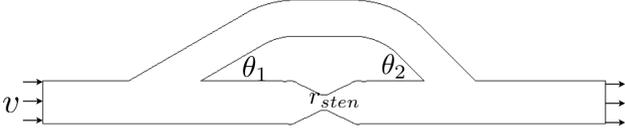


Fig. 1. Schematic of the bypass graft surgery with the two attachment angles, inlet velocity and stenosis radius shown.

We assume that the area of low WSS regions is fully determined by the actual anastomosis angles θ , the stenosis radius r , and the inflow velocity v . Given θ and $\omega = (r, v)$, we denote by $f(\theta, \omega)$ the corresponding area of low WSS. We can use simulation to evaluate $f(\theta, \omega)$ exactly. However, each evaluation is time-consuming, limiting how many times we may perform this evaluation.

To optionally include risk aversion into our objective function, we define a utility function U by

$$U(\theta, \omega) = -f(\theta, \omega) \quad \text{or} \quad U(\theta, \omega) = e^{-\alpha f(\theta, \omega)},$$

where $\alpha > 0$ is a parameter that models aversion to risk, with larger values of α corresponding to more aversion to risk. The second definition can be used to control the standard deviation (sensitivity) of f due to input uncertainties.

For analytical convenience, we suppose that our probability distributions over ω and δ are independent and normal (ω may be truncated at 0). Denote by $p(\delta, \omega)$ their joint pdf, which is assumed known.

Our overarching goal is to find the target anastomosis angles x that maximize the expected value of $U(\cdot, \cdot)$, i.e., we want to solve

$$\max_x g(x), \quad (1)$$

where

$$g(x) := \iint U(x + \delta, \omega) p(\delta, \omega) d\delta d\omega \quad (2)$$

is the expected utility that results from using target values x .

III. STATISTICAL INFERENCE AND VALUE OF INFORMATION ANALYSIS

To support the solution to the optimization problem, we use Bayesian statistics to provide an estimate of $U(\theta, \omega)$ across *all* points (θ, ω) , based on those points at which U has actually been evaluated. This statistical framework also provides uncertainties associated with these estimates. This is useful because evaluating U is time-consuming, and so we cannot simply evaluate it at each point of interest. These estimates of U , and their associated uncertainties, then imply

estimates and uncertainties of $g(x)$ across the domain of x . In this section, we first describe the statistical framework in which this estimation takes place. We then describe a value of information analysis based upon this statistical framework, in which we quantify the value of evaluating U at a given set of previously unevaluated values. This quantification of the value of information will then be used later in Section IV to create an algorithm for solving (1).

We work in a Bayesian framework, in which we place a Gaussian process (GP) prior distribution over the function L . For an overview of GP priors see [9].

$$U(\cdot, \cdot) \sim \text{GP}(\mu_0(\cdot, \cdot), \Sigma_0(\cdot, \cdot, \cdot, \cdot)),$$

where

$$\begin{aligned} \mu_0: (\theta, \omega) &\mapsto \mathbb{R}, \\ \Sigma_0: (\theta, \omega, \theta', \omega') &\mapsto \mathbb{R}, \end{aligned}$$

and Σ_0 is a positive semi-definite function. A typical choice of Σ_0 is the square exponential covariance function (see Section V-A). At each time $n = 1, 2, \dots$, our algorithm will evaluate some point (θ_n, ω_n) , and observe the resulting objective, $y_n = U(\theta_n, \omega_n)$. Define $D_n = \{\theta_{1:n}, \omega_{1:n}, y_{1:n}\}$ to contain all of this data. The posterior distribution of U at time n is then

$$U(\cdot, \cdot) | D_n \sim \text{GP}(\mu_n(\cdot, \cdot), \Sigma_n(\cdot, \cdot, \cdot, \cdot)),$$

where μ_n and Σ_n can be computed using standard results from Bayesian linear regression (see, e.g., [9] or [19]). Section V-B gives explicit expressions for μ_n and Σ_n .

Denote by \mathbb{E}_n and Cov_n the expectation and covariance conditioned on D_n , respectively. That is, \mathbb{E}_n and Cov_n are the expectation and covariance under the posterior distribution at time n . Then, we relate the posterior distribution on U to the posterior distribution on the function g via the following expressions. First, the posterior mean of the function g at an arbitrary point x can be calculated by interchanging integration over the values of $g(x)$ with the integration defining $g(x)$ in (2) via Fubini's theorem to obtain,

$$\mathbb{E}_n[g(x)] = \iint \mu_n(x + \delta, \omega) p(\delta, \omega) d\delta d\omega, \quad (3)$$

A similar computation provides the covariance between $g(x)$ and $g(x')$ at two arbitrary points x and x' in the following expression.

$$\begin{aligned} &\text{Cov}_n[g(x), g(x')] \\ &= \iiint \Sigma_n(x + \delta, \omega, x' + \delta', \omega') p(\delta, \omega) p(\delta', \omega') d\delta d\omega d\delta' d\omega'. \end{aligned} \quad (4)$$

Note that taking $x = x'$ gives an expression for the variance.

We will frequently refer to the posterior mean of $g(x)$, and so for brevity we introduce the notation

$$a_n(x) = \mathbb{E}_n[g(x)], \quad (5)$$

which is defined in terms of $\mu_n(\cdot, \cdot)$ by (3). Section V-C gives an explicit expression for $a_n(x)$. Then, if we were stop after

n evaluations of the simulator and choose the solution to (1) with the best estimated value, we would choose

$$x_n^* = \operatorname{argmax}_x \mathbb{E}_n [g(x)] = \operatorname{argmax}_x a_n(x).$$

In a formal sense, this solution is Bayes-optimal when we are neutral with respect to the risk introduced by our uncertainty about the simulation's output.

We now conduct an analysis to determine the expected solution quality that will result from a single additional evaluation of the simulator. The improvement in solution quality is then the value of the information provided by this additional evaluation.

Consider a given time n , and a given candidate point (θ, ω) to evaluate at time $n+1$. The expected quality of the best solution we can obtain *after* we observe the sample $y_{n+1} = U(\theta, \omega)$ that results from this evaluation is $\max_x a_{n+1}(x)$. This quantity is unknown at time n , as it depends on the outcome y_{n+1} . If we calculate its expected value at time n , and take the difference between this expected solution quality and the solution quality $\max_x a_n(x)$ that we have at time n , then we obtain the value of the information achieved from measuring (θ, ω) at time $n+1$,

$$V_n(\theta, \omega) = \mathbb{E}_n \left[\max_x a_{n+1}(x) \mid \theta_{n+1} = \theta, \omega_{n+1} = \omega \right] - \max_x a_n(x).$$

The algorithm we present in §IV seeks to evaluate the simulator at the point maximizing the value of information. That is, we want to evaluate at time $n+1$

$$(\theta_{n+1}, \omega_{n+1}) = \operatorname{argmax}_{\theta, \omega} V_n(\theta, \omega). \quad (6)$$

We now show how to compute $V_n(\theta, \omega)$. To perform this computation, we must first determine the distribution of $a_{n+1}(x)$ conditioned on D_n and $(\theta_{n+1}, \omega_{n+1})$ for an arbitrary x . The following lemma describes this distribution.

Lemma 1: Define

$$\begin{aligned} & b_n(x, \theta_{n+1}, \omega_{n+1}) \\ &= \left[\iiint \left[\Sigma_n(x+\delta, \omega, x+\delta', \omega') - \Sigma_{n+1}(x+\delta, \omega, x+\delta', \omega') \right] \right. \\ & \quad \left. p(\delta, \omega) p(\delta', \omega') d\delta d\omega d\delta' d\omega' \right]^{1/2}. \end{aligned} \quad (7)$$

Then

$$a_{n+1}(x) \mid D_n, \theta_{n+1}, \omega_{n+1} \sim \mathcal{N}(a_n(x), b_n^2(x, \theta_{n+1}, \omega_{n+1})). \quad (8)$$

Section V-C gives an explicit expression for $b_n(x, \theta_{n+1}, \omega_{n+1})$. Denote by \mathcal{X} the set of design variables x under consideration. We assume that \mathcal{X} is discrete and finite. Define the following two vectors

$$\begin{aligned} \vec{a}_n &= \{a_n(x) : x \in \mathcal{X}\}, \\ \vec{b}_n(\theta, \omega) &= \{b_n(x, \theta_{n+1}, \omega_{n+1}) \mid \theta_{n+1} = \theta, \omega_{n+1} = \omega : x \in \mathcal{X}\}. \end{aligned} \quad (9)$$

Then

$$V_n(\theta, \omega) = h(\vec{a}_n, \vec{b}_n(\theta, \omega)), \quad (10)$$

where

$$h(\vec{a}, \vec{b}) := \mathbb{E} \left[\max_i a_i + b_i Z \right] - \max_i a_i.$$

and Z is a standard normal variable. [11] gives an algorithm for computing h and [20] provides a MATLAB implementation. The derivative of $V_n(\theta, \omega)$ with respect to θ and ω , denoted by $\nabla_\theta V_n(\theta, \omega)$ and $\nabla_\omega V_n(\theta, \omega)$, is also available, and is described in Section V-D. We can then solve (6) using multi-start gradient ascent.

IV. ALGORITHM

We now summarize the algorithm that implements this value of information approach.

- 1) Evaluate U at a number of randomly chosen (θ, ω) . Fit a GP prior (see Section V-A) to U based on these evaluations, using maximum likelihood estimation.
- 2) At each time $n \geq 0$:
 - a) If the stopping rule is met, go to Step 4; else go to Step 2b.
 - b) Update $\vec{a}_n, \vec{b}_n(\cdot, \cdot), V_n(\cdot, \cdot)$ and $\nabla V_n(\cdot, \cdot)$ according to (9), (10) and Section V-C, V-D.
 - c) Maximize $V_n(\cdot, \cdot)$ using multi-start gradient ascent. Let $(\theta_{n+1}, \omega_{n+1})$ be the minimizer, and evaluate $U(\theta_{n+1}, \omega_{n+1})$.
- 3) Increase n and return to Step 2.
- 4) Report $x_n^* = \operatorname{argmax}_x a_n(x)$ as our final solution.

V. DETAILED COMPUTATIONS

In this section, we provide explicit expressions for the quantities introduced in the previous sections. We first describe the GP model in Section V-A, and then compute $\mu_n(\cdot, \cdot), \Sigma_n(\cdot, \cdot, \cdot, \cdot)$ in Section V-B, $a_n(\cdot), b_n(\cdot, \theta_{n+1}, \omega_{n+1})$ in Section V-C, and $\nabla V_n(\cdot, \cdot)$ in Section V-D.

A. Gaussian process priors

GP priors are frequently used in the Bayesian global optimization literature [21], [22], [15], where people use such priors to model their belief about an implicit continuous function over \mathbb{R}^d that closer arguments are more likely to correspond to similar values.

The previous work has demonstrated that the correlations in a GP prior are extremely important for reducing the number of samples needed to evaluate an expensive function, because they allow us to learn about areas that have not been measured from those that have.

In our particular GP prior for U , the covariance between $U(\theta, \omega)$ and $U(\theta', \omega')$ for some

$$\theta = \begin{bmatrix} \theta^{(1)} \\ \vdots \\ \theta^{(d_1)} \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega^{(1)} \\ \vdots \\ \omega^{(d_2)} \end{bmatrix}, \quad \theta' = \begin{bmatrix} \theta'^{(1)} \\ \vdots \\ \theta'^{(d_1)} \end{bmatrix}, \quad \omega' = \begin{bmatrix} \omega'^{(1)} \\ \vdots \\ \omega'^{(d_2)} \end{bmatrix}$$

($d_1 = d_2 = 2$), i.e. $\Sigma_0(\theta, \omega, \theta', \omega')$, is a decreasing function of the distance between (θ, ω) and (θ', ω') . In this work, we

use the following square exponential covariance function:

$$\begin{aligned} & \Sigma_0(\theta, \omega, \theta', \omega') \\ &= \sigma_0^2 \cdot \exp\left(-\sum_{k=1}^{d_1} \alpha_1^{(k)} [\theta^{(k)} - \theta'^{(k)}]^2 - \sum_{k=1}^{d_2} \alpha_2^{(k)} [\omega^{(k)} - \omega'^{(k)}]^2\right), \end{aligned} \quad (11)$$

where σ_0^2 is the common prior variance, and $\alpha_1^{(1)}, \dots, \alpha_1^{(d_1)}, \alpha_2^{(1)}, \dots, \alpha_2^{(d_2)}$ are the length scales. Values of these parameters are usually obtained using maximum likelihood estimation from the observations of U . This and other commonly used covariance functions, e.g., the Matern covariance function, are carefully discussed in [9] Section 4.

The mean of a GP prior is usually a linear regression function. Typical choices for $\mu_0(\cdot, \cdot)$ include

- 1) zero order polynomial (constant): $\mu_0(\theta, \omega) \equiv \xi$,
- 2) first order polynomial (linear):

$$\mu_0(\theta, \omega) = \sum_{k=1}^{d_1} \xi_1^{(k)} \theta^{(k)} + \sum_{k=1}^{d_2} \xi_2^{(k)} \omega^{(k)},$$

- 3) second order polynomial (quadratic), etc.,

where $\xi, \xi_1^{(1)}, \dots, \xi_1^{(d_1)}, \xi_2^{(1)}, \dots, \xi_2^{(d_2)}$ are the coefficients of the polynomials (“basis functions”). We use the generalized least squares estimates of these coefficients in practice (see [23] or [9] Section 2, 5).

To validate the GP model for our bypass graft surgery application, we apply leave-one-out cross-validation of the model with different covariance functions and regression functions using 137 observations from the cardiovascular simulation. As an example, Figure 2 shows the validation results of a GP prior with covariance (11) and a constant mean. We see that this model fits the data sufficiently well except for a very small number of outliers.

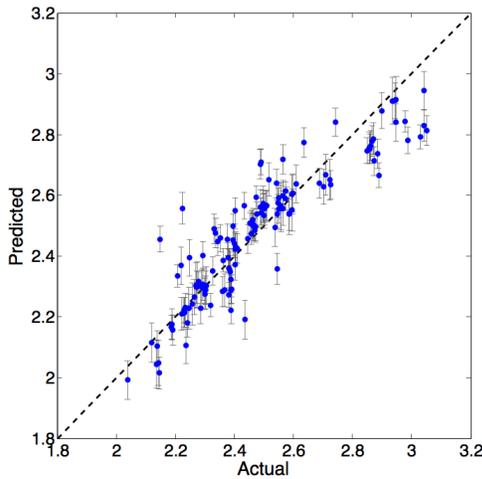


Fig. 2. Leave-one-out cross-validation of the Gaussian process prior with covariance (11) and a constant mean, using 137 observations from the cardiovascular simulation. Each dot compares the actual value of an observation against its predicted value from the other observations. Each error bar is the 95%-confidence interval of the corresponding prediction.

B. $\mu_n(\cdot, \cdot)$ and $\Sigma_n(\cdot, \cdot, \cdot, \cdot)$

We briefly describe the GP posterior distribution of U in this subsection. Define

$$\tilde{Y} = \begin{bmatrix} y_1 - \mu_0(\theta_1, \omega_1) \\ \vdots \\ y_n - \mu_0(\theta_n, \omega_n) \end{bmatrix}, T_n = \begin{bmatrix} \Sigma_0(\theta_1, \omega_1, \theta_1, \omega_1) & \cdots & \Sigma_0(\theta_1, \omega_1, \theta_n, \omega_n) \\ \vdots & \ddots & \vdots \\ \Sigma_0(\theta_n, \omega_n, \theta_1, \omega_1) & \cdots & \Sigma_0(\theta_n, \omega_n, \theta_n, \omega_n) \end{bmatrix}, \quad (12)$$

and

$$t_n(\cdot, \cdot) = [\Sigma_0(\cdot, \cdot, \theta_1, \omega_1) \cdots \Sigma_0(\cdot, \cdot, \theta_n, \omega_n)] T_n^{-1}. \quad (13)$$

Then for arbitrary θ, ω and θ', ω' , by the Kalman filter equations (see, e.g., [19] Section 14.6), we have

$$\mu_n(\theta, \omega) = \mu_0(\theta, \omega) + t_n(\theta, \omega) \tilde{Y}, \quad (14)$$

$$\Sigma_n(\theta, \omega, \theta', \omega') = \Sigma_0(\theta, \omega, \theta', \omega') - t_n(\theta, \omega) \begin{bmatrix} \Sigma_0(\theta', \omega', \theta_1, \omega_1) \\ \vdots \\ \Sigma_0(\theta', \omega', \theta_n, \omega_n) \end{bmatrix}. \quad (15)$$

C. $a_n(\cdot)$ and $b_n(\cdot, \theta_{n+1}, \omega_{n+1})$

Based on Section V-A and V-B, we now explicitly compute $a_n(\cdot)$ and $b_n(\cdot, \theta_{n+1}, \omega_{n+1})$, which can then support the calculation of the value of information (10).

Suppose that

$$\delta^{(k)} \sim \mathcal{N}(\mu_1^{(k)}, 1/\beta_1^{(k)}), \quad k = 1, \dots, d_1,$$

$$\omega^{(k)} \sim \mathcal{N}(\mu_2^{(k)}, 1/\beta_2^{(k)}), \quad \omega^{(k)} \geq 0, \quad k = 1, \dots, d_2,$$

and that $\delta^{(1)}, \dots, \delta^{(d_1)}, \omega^{(1)}, \dots, \omega^{(d_2)}$ are mutually independent.

Define

$$S_0(x) = \iint \mu_0(x + \delta, \omega) p(\delta, \omega) d\delta d\omega, \quad (16)$$

and for $i = 1, \dots, n+1$,

$$S_i(x) = \iint \Sigma_0(x + \delta, \omega, \theta_i, \omega_i) p(\delta, \omega) d\delta d\omega. \quad (17)$$

Then by (5), (3) and (12)-(17), we have

$$\begin{aligned} a_n(x) &= S_0(x) + \iint t_n(x + \delta, \omega) \tilde{Y} p(\delta, \omega) d\delta d\omega \\ &= S_0(x) + [S_1(x) \cdots S_n(x)] T_n^{-1} \tilde{Y}. \end{aligned} \quad (18)$$

By (15) and the Sherman-Morrison-Woodbury formula (see, e.g., [24]), we can write

$$\begin{aligned} & \Sigma_n(\theta, \omega, \theta', \omega') - \Sigma_{n+1}(\theta, \omega, \theta', \omega') \\ &= \frac{\Sigma_n(\theta, \omega, \theta_{n+1}, \omega_{n+1}) \Sigma_n(\theta', \omega', \theta_{n+1}, \omega_{n+1})}{\Sigma_n(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1})}. \end{aligned}$$

Plug this and (12)-(17) into (7), then we have

$$\begin{aligned} b_n(x, \theta_{n+1}, \omega_{n+1}) &= \frac{\iint \Sigma_n(x + \delta, \omega, \theta_{n+1}, \omega_{n+1}) p(\delta, \omega) d\delta d\omega}{\sqrt{\Sigma_n(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1})}} \\ &= \frac{S_{n+1}(x) - [S_1(x) \cdots S_n(x)] T_n^{-1} \alpha}{\sqrt{\Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - \alpha^T T_n^{-1} \alpha}}, \end{aligned} \quad (19)$$

where

$$\alpha = \begin{bmatrix} \Sigma_0(\theta_1, \omega_1, \theta_{n+1}, \omega_{n+1}) \\ \vdots \\ \Sigma_0(\theta_n, \omega_n, \theta_{n+1}, \omega_{n+1}) \end{bmatrix}$$

and T denotes matrix transposition.

Given a GP prior with covariance (11) and a constant mean ξ , we now give an explicit expression for $S_i(x)$ ($i = 0, 1, \dots, n+1$). By plugging in (11) and $p(\delta, \omega)$, we have

$$S_i(x) = \sigma_0^2 \cdot \prod_{k=1}^{d_1} \left[\sqrt{\frac{\beta_1^{(k)}}{2\pi}} \cdot A_i^{(k)} \right] \cdot \prod_{k=1}^{d_2} \left[\sqrt{\frac{\beta_2^{(k)}}{2\pi}} \cdot B_i^{(k)} \right]$$

for $i = 1, \dots, n+1$, and $S_0(x) = \xi$, where

$$A_i^{(k)} = \int_{-\infty}^{\infty} \exp\left(-\alpha_1^{(k)} [x^{(k)} + \delta^{(k)} - \theta_i^{(k)}]^2 - \frac{\beta_1^{(k)}}{2} [\delta^{(k)} - \mu_1^{(k)}]^2\right) d\delta^{(k)},$$

$$B_i^{(k)} = \int_0^{\infty} \exp\left(-\alpha_2^{(k)} [\omega^{(k)} - \omega_i^{(k)}]^2 - \frac{\beta_2^{(k)}}{2} [\omega^{(k)} - \mu_2^{(k)}]^2\right) d\omega^{(k)}.$$

Simple algebra then yields

$$A_i^{(k)} = \sqrt{\frac{\pi}{\alpha_1^{(k)} + \frac{1}{2}\beta_1^{(k)}}} \cdot \exp\left[-\alpha_1^{(k)} [x^{(k)} - \theta_i^{(k)}]^2 - \frac{\beta_1^{(k)} [\mu_1^{(k)}]^2}{2} + \frac{[\beta_1^{(k)} \mu_1^{(k)} + 2\alpha_1^{(k)} [x^{(k)} - \theta_i^{(k)}]^2]}{4\alpha_1^{(k)} + 2\beta_1^{(k)}}\right]$$

and

$$B_i^{(k)} = \frac{\Phi\left(\frac{\beta_2^{(k)} \mu_2^{(k)} + 2\alpha_2^{(k)} \omega_i^{(k)}}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}}\right)}{\sqrt{\pi^{-1} \left[\alpha_2^{(k)} + \frac{1}{2}\beta_2^{(k)}\right]}} \cdot \exp\left[-\alpha_2^{(k)} [\omega_i^{(k)}]^2 - \frac{1}{2}\beta_2^{(k)} [\mu_2^{(k)}]^2 + \frac{[\beta_2^{(k)} \mu_2^{(k)} + 2\alpha_2^{(k)} \omega_i^{(k)}]^2}{4\alpha_2^{(k)} + 2\beta_2^{(k)}}\right],$$

where Φ is the standard normal cdf.

D. $\nabla V_n(\cdot, \cdot)$

We briefly describe the algorithm in [11] for computing h here to provide notation and context that supports the computation of $\nabla_{\theta} V_n(\theta, \omega)$ and $\nabla_{\omega} V_n(\theta, \omega)$.

First, $h(\vec{a}, \vec{b})$ does not change if we reorder the components of the inputs. Thus, without loss of generality, we assume that the b_i are in non-decreasing order and ties in b are broken so that $a_i \leq a_{i+1}$ if $b_i = b_{i+1}$. Then, we remove all those entries i for which $a_i + b_i z < \max_{j \neq i} a_j + b_j z$ for all values of z . An algorithm for doing this is given in Algorithm 1 in [25]. This gives new vectors \vec{a}' and \vec{b}' with $|\vec{a}'| = |\vec{b}'| \leq |\vec{a}| = |\vec{b}|$, where $|\cdot|$ denotes the length of a vector. Then,

$$h(\vec{a}, \vec{b}) = \sum_{i=1}^{|\vec{a}'|-1} (b'_{i+1} - b'_i) f(-|c_i|), \quad (20)$$

where

$$f(-z) := \varphi(z) - z\Phi(-z),$$

$$c_i := -\frac{a'_{i+1} - a'_i}{b'_{i+1} - b'_i} \quad \text{for } i = 1, \dots, |\vec{a}'| - 1, \quad (21)$$

and φ and Φ are the standard normal pdf and cdf.

Now let \vec{a}' and \vec{b}' be the reordering of \vec{a}_n and $\vec{b}_n(\theta, \omega)$ respectively in the acceptance set of Algorithm 1 in [25]. Then if $|\vec{a}'| = 1$, $V_n(\theta, \omega) = h(\vec{a}_n, \vec{b}_n(\theta, \omega)) = 0$, and $\nabla V_n(\theta, \omega) = \vec{0}$. Otherwise,

$$\begin{aligned} \nabla V_n(\theta, \omega) &= -\nabla h(\vec{a}_n, \vec{b}_n(\theta, \omega)) \\ &= \sum_{i=1}^{|\vec{a}'|-1} (b'_{i+1} - b'_i) \Phi(-|c_i|) \nabla |c_i| - (\nabla b'_{i+1} - \nabla b'_i) f(-|c_i|) \quad (22) \\ &= \sum_{i=1}^{|\vec{a}'|-1} (\nabla b'_{i+1} - \nabla b'_i) [-|c_i| \Phi(-|c_i|) - f(-|c_i|)] \quad (23) \\ &= \sum_{i=1}^{|\vec{a}'|-1} (\nabla b'_i - \nabla b'_{i+1}) \varphi(|c_i|), \end{aligned}$$

where (22) follows from (20) and $\nabla f = \Phi$; (23) follows since $\nabla a'_i = 0$ for all i , and by the definition in (21),

$$\nabla |c_i| = \frac{-|a'_{i+1} - a'_i| (\nabla b'_{i+1} - \nabla b'_i)}{(b'_{i+1} - b'_i)^2}.$$

It then suffices to compute $\nabla b'_i$ for all i , or equivalently, $\nabla_{\theta_{n+1}} b_n(x, \theta_{n+1}, \omega_{n+1})$ and $\nabla_{\omega_{n+1}} b_n(x, \theta_{n+1}, \omega_{n+1})$ for all x . Now let ∇ denote the gradient w.r.t. θ_{n+1} or ω_{n+1} . By (19), it is clear that

$$\begin{aligned} \nabla b_n(x, \theta_{n+1}, \omega_{n+1}) &= \gamma_1 \left(\nabla S_{n+1}(x) - \nabla(\alpha^T) T_n^{-1} \begin{bmatrix} S_1(x) \\ \vdots \\ S_n(x) \end{bmatrix} \right) \\ &\quad - \frac{1}{2} \gamma_1^3 \gamma_2 [\nabla \Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - 2\nabla(\alpha^T) T_n^{-1} \alpha], \end{aligned} \quad (24)$$

where

$$\begin{aligned} \gamma_1 &= [\Sigma_0(\theta_{n+1}, \omega_{n+1}, \theta_{n+1}, \omega_{n+1}) - \alpha^T T_n^{-1} \alpha]^{-1/2}, \\ \gamma_2 &= S_{n+1}(x) - [S_1(x) \ \cdots \ S_n(x)] T_n^{-1} \alpha, \\ \nabla(\alpha^T) &= [\nabla \Sigma_0(\theta_1, \omega_1, \theta_{n+1}, \omega_{n+1}) \ \cdots \ \nabla \Sigma_0(\theta_n, \omega_n, \theta_{n+1}, \omega_{n+1})]. \end{aligned}$$

With a GP prior (11), we can write (24) explicitly by plugging in

$$\begin{aligned} \nabla_{\theta_{n+1}} \Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}) &= \begin{cases} \vec{0}, & i=n+1, \\ 2\alpha_1(\theta_i - \theta_{n+1}) \Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}), & i=1, \dots, n, \end{cases} \\ \nabla_{\omega_{n+1}} \Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}) &= \begin{cases} \vec{0}, & i=n+1, \\ 2\alpha_2(\omega_i - \omega_{n+1}) \Sigma_0(\theta_i, \omega_i, \theta_{n+1}, \omega_{n+1}), & i=1, \dots, n, \end{cases} \end{aligned}$$

and

$$\nabla_{\theta_{n+1}^{(k)}} S_{n+1}(x) = 2\alpha_1^{(k)} S_{n+1}(x) \left[x^{(k)} - \theta_{n+1}^{(k)} - v_1 \right],$$

$$\nabla_{\omega_{n+1}^{(k)}} S_{n+1}(x) = 2\alpha_2^{(k)} S_{n+1}(x) \left[\frac{\varphi(v_2)/\Phi(v_2) + v_2}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}} - \omega_{n+1}^{(k)} \right],$$

where

$$v_1 = \frac{\beta_1^{(k)} \mu_1^{(k)} + 2\alpha_1^{(k)} (x^{(k)} - \theta_{n+1}^{(k)})}{2\alpha_1^{(k)} + \beta_1^{(k)}}, v_2 = \frac{\beta_2^{(k)} \mu_2^{(k)} + 2\alpha_2^{(k)} \omega_{n+1}^{(k)}}{\sqrt{2\alpha_2^{(k)} + \beta_2^{(k)}}}.$$

VI. NUMERICAL RESULTS

We now explore the performance of the sampling algorithm proposed in Section IV.

We consider an initial test problem, where we simplify the design variable x and the environmental variable ω to be one-dimensional, and assume no implementation error ($\delta \equiv 0$). Suppose that $\omega \sim \mathcal{N}(1, 1/9)$ and that the deterministic utility function is

$$U = -100(\omega - \theta)^2 - (1 - \theta)^2. \quad (25)$$

We compare our algorithm against the random search algorithm, in which each (θ_n, ω_n) is selected independently and uniformly at random. In each sample path, we fit a GP prior distribution with covariance (11) and a constant mean to U after 10 initial random evaluations, and re-fit it after 40 evaluations. We measure the performance of the algorithms by their expected opportunity cost $\mathbb{E}[\max_x g(x) - g(x_n^*)]$ at each time n , where in this problem the objective function g has an analytical form.

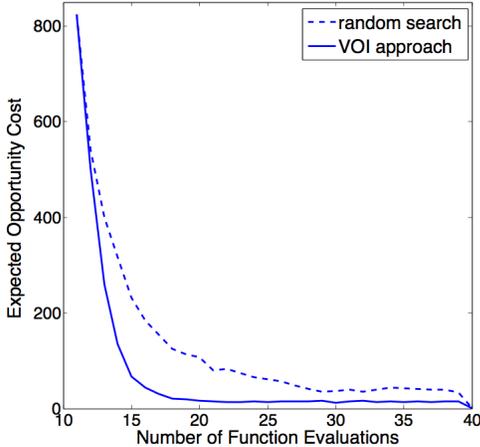


Fig. 3. Performance of the proposed value-of-information (VOI) based algorithm and the random search algorithm in the test problem (25).

Figure 3 shows that the proposed algorithm significantly outperforms the random search algorithm, before they both achieve near-optimal solutions after the GP model re-fitting. The results are based on 500 sample paths for each algorithm.

For more sophisticated problems, the proposed value-of-information approach will demonstrate greater advantage over the naive strategies. We will include later experimental

results for other test problems and the graft surgery application.

APPENDIX

Proof of Lemma 1

Proof: By (3) and (5),

$$a_{n+1}(x) = \int \mu_{n+1}(x + \delta, \omega) p(\delta, \omega) d\delta d\omega.$$

Since the posterior mean $\mu_{n+1}(\cdot, \cdot)$ is a linear function of the observations up to time n and y_{n+1} , we can write $a_{n+1}(x) | D_n, \theta_{n+1}, \omega_{n+1}$ as

$$s_n(x, \theta_{n+1}, \omega_{n+1}) + t_n(x, \theta_{n+1}, \omega_{n+1}) \cdot y_{n+1},$$

where s_n and t_n are real-valued, deterministic functions of D_n . Now since y_{n+1} conditioned on $D_n, \theta_{n+1}, \omega_{n+1}$ is normally distributed, we know that $a_{n+1}(x)$ is also normally distributed conditioned on $D_n, \theta_{n+1}, \omega_{n+1}$.

By the tower property,

$$a_n(x) = \mathbb{E}_n[g(x)] = \mathbb{E}_n[\mathbb{E}_{n+1}[g(x)]].$$

Also since Σ_{n+1} does not depend on y_{n+1} (it is fully determined by D_n, θ_{n+1} and ω_{n+1}), we know that b_n is well defined. By (4) and the conditional variance formula,

$$\begin{aligned} b_n(x, \theta_{n+1}, \omega_{n+1}) &= \text{Var}_n[g(x)] - \text{Var}_{n+1}[g(x)] \\ &= \text{Var}_n[g(x)] - \mathbb{E}_n[\text{Var}_{n+1}[g(x)] | \theta_{n+1}, \omega_{n+1}] \\ &= \text{Var}_n[\mathbb{E}_{n+1}[g(x)] | \theta_{n+1}, \omega_{n+1}]. \end{aligned}$$

Thus (8) follows. ■

REFERENCES

- [1] A. Booker, J. Dennis, P. Frank, D. Serafini, V. Torczon, and M. Trosset, "A rigorous framework for optimization of expensive functions by surrogates," *Structural and Multidisciplinary Optimization*, vol. 17, no. 1, pp. 1–13, 1999.
- [2] A. Marsden, M. Wang, J. Dennis, and P. Moin, "Optimal aeroacoustic shape design using the surrogate management framework," *Optimization and Engineering*, vol. 5, no. 2, pp. 235–262, 2004.
- [3] S. Sankaran and A. Marsden, "The impact of uncertainty on shape optimization of idealized bypass graft models in unsteady flow," *Physics of Fluids*, vol. 22, p. 121902, 2010.
- [4] —, "A stochastic collocation method for uncertainty quantification in cardiovascular simulations," *Journal of Biomechanical Engineering*, vol. 133, p. 031001, 2011.
- [5] P. Chang, B. Williams, T. Santner, W. Notz, and B. D.L., "Robust optimization of total joint replacements incorporating environmental variables," *Journal of Biomechanical Engineering*, vol. 121, pp. 304–310, 1999.
- [6] T. Santner, W. Notz, and B. D.L., "Robust design and analysis of total joint replacements: Finite element model experiments with environmental variables," *Journal of Biomechanical Engineering*, vol. 123, pp. 239–246, 2001.
- [7] A. O'Hagan, "Bayes–hermite quadrature," *Journal of Statistical Planning and Inference*, vol. 29, no. 3, pp. 245–260, 1991.
- [8] C. Rasmussen and Z. Ghahramani, "Bayesian monte carlo," *Advances in neural information processing systems*, vol. 15, pp. 489–496, 2003.
- [9] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006. [Online]. Available: <http://www.gaussianprocess.org/gpml>
- [10] S. Gupta and K. Miescke, "Bayesian look ahead one-stage sampling allocations for selection of the best population," *Journal of statistical planning and inference*, vol. 54, no. 2, pp. 229–244, 1996.
- [11] P. Frazier, W. B. Powell, and S. Dayanik, "A knowledge gradient policy for sequential information collection," *SIAM J. on Control and Optimization*, vol. 47, no. 5, pp. 2410–2439, 2008.

- [12] S. Chick and N. Gans, "Economic analysis of simulation selection problems," *Management Sci.*, vol. 55, no. 3, pp. 421–437, 2009.
- [13] S. Chick and P. Frazier, "The conjunction of the knowledge gradient and economic approach to simulation selection," in *Winter Simulation Conference Proceedings, 2009*. IEEE, 2009, pp. 528–539.
- [14] R. Bechhofer, T. Santner, and D. Goldsman, *Design and Analysis of Experiments for Statistical Selection, Screening and Multiple Comparisons*. New York: J.Wiley & Sons, 1995.
- [15] D. Jones, M. Schonlau, and W. Welch, "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, 1998.
- [16] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans, "Automatic gait optimization with gaussian process regression," in *Proc. of IJCAI, 2007*, pp. 944–949.
- [17] D. Ginsbourger, R. Le Riche, and L. Carraro, "A multi-points criterion for deterministic parallel global optimization based on Gaussian processes," 2008.
- [18] P. Frazier, "Knowledge-gradient methods for statistical learning," Ph.D. dissertation, Princeton University, 2009.
- [19] A. Gelman, J. Carlin, H. Stern, and D. Rubin, *Bayesian data analysis*, 2nd ed. Boca Raton, FL: CRC Press, 2004.
- [20] P. Frazier, <http://people.orie.cornell.edu/pfrazier/src.html>, 2009–2010.
- [21] H. J. Kushner, "A new method of locating the maximum of an arbitrary multi-peak curve in the presence of noise," *Journal of Basic Engineering*, vol. 86, pp. 97–106, 1964.
- [22] J. Mockus, *Bayesian approach to global optimization: theory and applications*. Kluwer Academic, Dordrecht, 1989.
- [23] D. Huang, T. Allen, W. Notz, and N. Zeng, "Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models," *Journal of Global Optimization*, vol. 34, no. 3, pp. 441–466, 2006.
- [24] G. Golub and C. Van Loan, *Matrix Computations*. Baltimore, MD: John Hopkins University Press, 1996.
- [25] P. Frazier, W. B. Powell, and S. Dayanik, "The knowledge gradient policy for correlated normal beliefs," *INFORMS Journal on Computing*, vol. 21, no. 4, pp. 599–613, 2009.