Statistica Sinica **14**(2004), 571-590

DESIGNING COMPUTER EXPERIMENTS TO DETERMINE ROBUST CONTROL VARIABLES

Jeffrey S. Lehman[1], Thomas J. Santner[2] and William I. Notz[2]

[1]*Battelle Memorial Institute and* [2]*The Ohio State University*

Abstract: This paper is concerned with the design of computer experiments when there are two types of inputs: control variables and environmental variables. Control variables, also called manufacturing variables, are determined by a product designer while environmental variables, called noise variables in the quality control literature, are uncontrolled in the field but take values that are characterized by a probability distribution. Our goal is to find a set of control variables at which the response is insensitive to the value of the environmental variables, a "robust" choice of control variables. Such a choice ensures that the mean response is as insensitive as possible to perturbations of the nominal environmental variable distribution. We present a sequential strategy to select the inputs at which to observe the response so as to determine a robust setting of the control variables. Our solution is Bayesian; the prior takes the response as a draw from a stationary Gaussian stochastic process. Given the previous information, the sequential algorithm computes for each untested site the "improvement" over the current guess of the optimal robust setting. The design selects the next site to maximize the expected improvement criterion.

Key words and phrases: Computer experiments, expected improvement, noise variables, robust control variables, robust optimization, sequential design.

1. Introduction

Computer experiments are a relatively new type of empirical investigation in the computer dependent world of the 21^{st} century. Historically, physical experiments were required to study processes in-vivo. The complexity of such systems has sometimes made physical experiments prohibitive, if not impossible, due to time constraints, physical constraints, or financial constraints. Complex systems that lend themselves to mathematical modeling can be studied via computer codes, i.e., by a computer model of the system. Such a computer code is able to compute *responses* at arbitrary inputs using numerical methods, for example by finite element models in many engineering applications, or by a simulation run to the point of (essentially) no simulation error. Thus we are able to perform an experiment of the process of interest by submitting arbitrary *inputs* to the code to obtain one or more *responses*. As examples, Sacks, Welch, Mitchell and Wynn

(1989) describe the use of computer codes to improve the quality of integrated circuits; Chang, Williams, Notz, Santner and Bartel (1999) study the design of an optimal hip prosthesis based on the implants' proximal bone stress shielding and its toggling motion; Haylock and O'Hagan (1996) use a computer model to describe the dispersion of radioactive iodine in the human body.

Two difficulties of using computer experiments to study the input−output relationship of a physical phenomenon are: the code can be very time consuming to run, and the input can be (very) high-dimensional. This has led to the use of stochastic process (random function) models as the basis for interpolating the response with the corresponding predictions based on a small training sample of computer runs. This approach is best regarded from the Bayesian perspective with the deterministic computer code being treated as a realization of a stochastic process prior whose properties are determined by the prior information about the computer code (see Sacks, Welch, Mitchell and Wynn (1989) and Koehler and Owen (1996)). Then the predictive interpolator is used in place of the computer code to investigate the input–output relationship. For example, Jones, Schonlau and Welch (1998) use such an interpolator to find the setting of the input variables that optimizes the output of the computer code.

This paper considers settings where both control variables and environmental variables are present. As one example, in the hip prosthesis problem of Chang, Williams, Notz, Santner and Bartel (1999), the control variables described the geometry of the prosthesis and the environmental variables described the (uncontrolled) patient bone material properties and patient activity. As a second example, Welch, Yu, Kang and Sacks (1990) investigated the clock skew in the design of a large scale integrated circuit where the control variables consisted of the widths of the six transistors in the circuit, and the environmental variables described the current driving capabilities of the transistors. In both of these examples, the researchers were interested in finding a set of "optimal" values for the control variables.

We propose sequential designs for determining a *robust* choice of control variables in a computer experiment that also has environmental variables. Section 2 reviews different methods of defining robustness and focuses on finding a set of control variables at which the response is "insensitive" to the value of the environmental variables. Section 3 presents our model for the response. Section 4 outlines a sequential expected improvement algorithm in the spirit of Jones, Schonlau and Welch (1998) or Williams, Santner and Notz (2000) for finding a robust control variable value while Section 5 illustrates the performance of these algorithms with examples that involve several different experimental goals. Finally, Section 6 discusses some computational considerations for these algorithms and areas for future research.

2. Concepts of Robustness

Let $y(\cdot)$ denote the computer code output and $\boldsymbol{x} = (\boldsymbol{x}_c, \boldsymbol{x}_e)$ denote the vector input, where $\boldsymbol{x}_c$ is the vector of control variables and $\boldsymbol{x}_e$ is the vector of environmental variables. Also let $F(\cdot)$ denote the (tentative) joint distribution of the environmental variables, $\boldsymbol{X}_e$.

If $F(\cdot)$ is *known*, we typically focus attention either on determining the distribution of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$ ("uncertainty analysis", see O'Hagan and Haylock (1997) or O'Hagan, Kennedy and Oakley (1999) for examples) *or* on determining some summary of this distribution such as its mean

$$\mu_F(\boldsymbol{x}_c) = E_F\{y(\boldsymbol{x}_c, \boldsymbol{X}_e)\} \tag{1}$$

(see, for example, Williams, Santner and Notz (2000)).

If $F(\cdot)$ is *unknown* (either completely or up to a finite vector of parameters) then even the summary quantity $\mu_F(\boldsymbol{x}_c)$ may not be meaningful if its value is "sensitive" to the assumed $F(\cdot)$. To illustrate, consider the response $y(\cdot)$ plotted in the upper left-hand panel of Figure 1; this $y(\cdot)$ depends on a single real-valued control variable x_c, $0 \le x_c \le 1$, and a single real-valued environmental variable x_e, $0 \le x_e \le 1$. The upper right-hand panel of Figure 1 displays four X_e density functions whose associated $\mu_F(x_c)$ are pictured in the bottom panels of the figure. Suppose that it is desired to find the x_c that minimizes $\mu_F(x_c)$; this minimizer depends on $F(\cdot)$. If the X_e distribution is #2 then $x_c^\star \approx 0.22$ minimizes $\mu_F(x_c)$, but if the X_e distribution is #4, then $x_c^\star \approx 0.82$ minimizes $\mu_F(x_c)$.

This paper proposes sequential designs for a computer experiment to determine a "robust" choice of control variables $\boldsymbol{x}_c$ using a Taguchi-like approach. Suppose interest lies in the *mean* $\mu_F(\boldsymbol{x}_c)$. If $y(\boldsymbol{x}_c, \boldsymbol{x}_e)$ is relatively "flat" in $\boldsymbol{x}_e$ for a *given* $\boldsymbol{x}_c$ value, then the mean of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$ will be relatively independent of the choice of $F(\cdot)$ (and thus be robust to misspecification of $F(\cdot)$). To apply this idea, suppose that small values of $\mu_F(\boldsymbol{x}_c)$ are desirable. Then a robust value of $\boldsymbol{x}_c$ minimizes $\mu_F(\boldsymbol{x}_c)$ among $\boldsymbol{x}_c$'s for which $y(\boldsymbol{x}_c, \boldsymbol{x}_e)$ is "flat" in $\boldsymbol{x}_e$. We quantify the flatness of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$ by

$$\sigma_G^2(\boldsymbol{x}_c) = \mathrm{Var}_G[y(\boldsymbol{x}_c, \boldsymbol{X}_e)], \tag{2}$$

where $G(\cdot)$ is a distribution on $\boldsymbol{X}_e$ selected by the user. For example, one could take $G(\cdot)$ to be a uniform distribution on $\boldsymbol{X}_e$ or even allow $G(\cdot)$ to vary over a set of distributions. Taking $G = F$, the bottom right panel of Figure 1 plots $\sigma_F^2(x_c)$ for each of the four $\boldsymbol{X}_e$ distributions. The variance $\sigma_F^2(x_c)$ varies greatly over the four $F(\cdot)$ distributions for x_c near 0.22 but is relatively invariant to $F(\cdot)$ for x_c near 0.85. This suggests $y(0.85, x_e)$ is flatter than $y(0.22, x_e)$, which can be verified visually in this simple example.

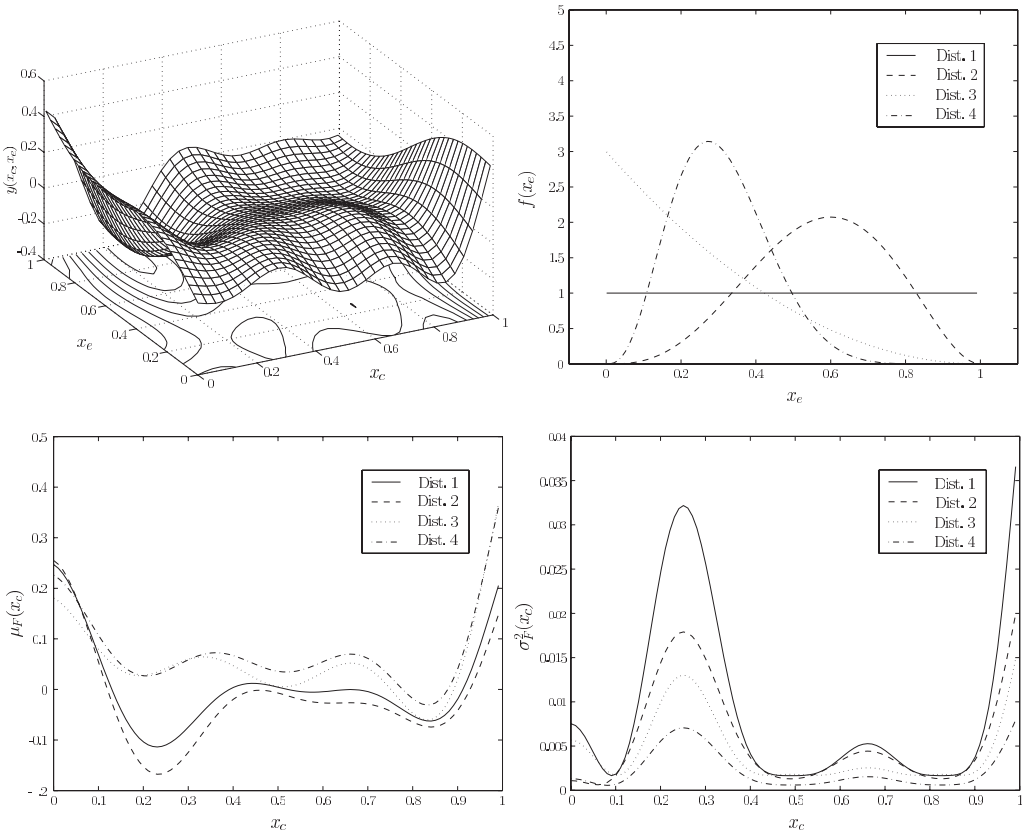

Figure 1. True $y(x_c, x_e)$ (top left panel), and true $\mu_F(x_c)$ (bottom left panel) and $\sigma_F^2(x_c)$ (bottom right panel) for four $\boldsymbol{X}_e$ distributions (top right panel).

Motivated by this observation, we define $\boldsymbol{x}_c^M$ to be *M-robust* if $\boldsymbol{x}_c^M$ minimizes $\mu_F(\boldsymbol{x}_c)$ subject to a constraint on $\sigma_G^2(\boldsymbol{x}_c)$. Alternatively, and perhaps more in keeping with the quality control concept of having a "target" mean, we define $\boldsymbol{x}_c^V$ to be *V-robust* if it minimizes $\sigma_G^2(\boldsymbol{x}_c)$ subject to a constraint on $\mu_F(\boldsymbol{x}_c)$. These formulations bear similarities to the constrained optimization problem for computer experiments with bivariate outputs where the interest is in optimizing one output, $y_1(\cdot)$, subject to input constraints defined by a second output, $y_2(\cdot)$. Schonlau, Welch and Jones (1998) and Williams, Lehman, Santner and Notz (2003) address this problem using stochastic process models for $(y_1(\cdot), y_2(\cdot))$ where the former assumed the $y_i(\cdot)$ come from independent Gaussian processes while the latter assumed the $Y_i(\cdot)$ were dependent so that, say, large values of one function could (stochastically) be associated with small values of the other. In contrast, here we consider two functions, $\mu_F(\cdot)$ and $\sigma_G^2(\cdot)$, defined for a *single* computer output and optimize one subject to constraints on the other. In Section 4, we propose sequential designs of computer experiments that can be used

to find M-robust and V-robust control variables. *For notational convenience, beginning with Section* 3 *the remainder of this paper will assume that* $G = F$ *but the algorithms proposed here can be modified in an obvious way if* $G \neq F$ *is desired in* (2).

Before sketching these algorithms we note briefly two other senses that the statistical literature has used to quantify the notion of a robust choice of control variables. When the goal is minimization of the response, the minimax approach to defining robustness Huber (1981) assumes that a family $\mathcal{G}$ of distributions can be specified that contains the unknown $F(\cdot)$. We define $\boldsymbol{x}_c^{\mathcal{G}}$ to be $\mathcal{G}$-*robust* if

$$\max_{G \in \mathcal{G}} \mu_G(\boldsymbol{x}_c^{\mathcal{G}}) = \min_{\boldsymbol{x}_c \in \mathcal{X}_c} \max_{G \in \mathcal{G}} \mu_G(\boldsymbol{x}_c).$$

Minimax robusness adopts a pessimistic viewpoint because it attempts to guard against the worst-case scenario among all $\boldsymbol{X}_e$ distributions in $\mathcal{G}$. A more Bayesian approach to robustness focuses on the mean

$$\mu^{\Pi}(\boldsymbol{x}_c) = \int_{G \in \mathcal{G}} \mu_G(\boldsymbol{x}_c) d\Pi(G), \tag{3}$$

over the possible $\boldsymbol{X}_e$ distributions in $\mathcal{G}$; here $\Pi(\cdot)$ is a prior distribution on $\mathcal{G}$. A $\boldsymbol{x}_c^{\Pi}$ that minimizes (3) is said to be Π-*robust*.

Both the minimax and Bayes notions of robustness require extra information −either the specification of the class $\mathcal{G}$ and possibly a prior $\Pi(\cdot)$. They also require the possibly difficult calculation of a *max* or *average* over this class. The Taguchi formulation of robustness has the practical advantages that it requires neither extra information nor a maximum/integral calculation.

3. Modeling

The approach adopted here is Bayesian. The prior model for the true response $y(\cdot)$ is hierarchical with first stage

$$Y(\boldsymbol{x}) = \boldsymbol{f}^{\top}(\boldsymbol{x})\boldsymbol{\beta} + Z(\boldsymbol{x}), \tag{4}$$

where the linear model $\boldsymbol{f}^{\top}(\cdot)\boldsymbol{\beta}$ represents the (nonstationary) global mean of the $Y(\cdot)$ process with $\boldsymbol{f}(\cdot)$ a k-vector of known regression functions and $\boldsymbol{\beta} \in \mathbb{R}^k$ a vector of unknown regression parameters, and $Z(\cdot)$ is a covariance stationary Gaussian stochastic process having mean zero, positive definite correlation function $R(\cdot)$, and unknown variance $\tau^2 > 0$. The specification of the model is completed by assuming the (non-informative) second-stage prior $[\boldsymbol{\beta}, \tau^2] \propto 1/\tau^2$ for the parameters $(\boldsymbol{\beta}, \tau^2)$, and that $R(\cdot) = R(\cdot|\boldsymbol{\gamma})$ is a known parametric correlation function. As would occur often in practice, the examples below allow $\boldsymbol{\gamma}$ to be unknown and set $\boldsymbol{\gamma}$ equal to its posterior mode wherever calculations involve known $\boldsymbol{\gamma}$.

Assume that the joint distribution of the environmental variables, $F(\cdot)$, is discrete on $\{\boldsymbol{x}_{e,j}\}_{j=1}^{n_e}$ with probabilities $P_F\{\boldsymbol{X}_e = \boldsymbol{x}_{e,j}\} = w_j$. Denote the mean and variance of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$ by

$$\mu_F(\boldsymbol{x}_c) = E_F\{y(\boldsymbol{x}_c, \boldsymbol{X}_e)\} = \sum_{j=1}^{n_e} w_j y(\boldsymbol{x}_c, \boldsymbol{x}_{e,j}) = \boldsymbol{w}^\top \boldsymbol{y}_{n_e}(\boldsymbol{x}_c), \tag{5}$$

$$\begin{aligned}\sigma_F^2(\boldsymbol{x}_c) = \mathrm{Var}_F[y(\boldsymbol{x}_c, \boldsymbol{X}_e)] &= \sum_{j=1}^{n_e} w_j (y(\boldsymbol{x}_c, \boldsymbol{x}_{e,j}) - \mu_F(\boldsymbol{x}_c))^2 \\ &= \boldsymbol{y}_{n_e}(\boldsymbol{x}_c)^\top A \boldsymbol{y}_{n_e}(\boldsymbol{x}_c),\end{aligned} \tag{6}$$

where $A = (I_{n_e} - \mathbf{1}_{n_e}\boldsymbol{w}^\top)^\top \mathrm{diag}(\boldsymbol{w})(I_{n_e} - \mathbf{1}_{n_e}\boldsymbol{w}^\top)$, $\boldsymbol{w} = (w_1, \ldots, w_{n_e})^\top$, $\boldsymbol{y}_{n_e}(\boldsymbol{x}_c)$ $= (y(\boldsymbol{x}_c, \boldsymbol{x}_{e,1}), \ldots, y(\boldsymbol{x}_c, \boldsymbol{x}_{e,n_e}))^\top$, I_n is the $n \times n$ identity matrix, $\mathbf{1}_n$ is an $n \times 1$ vector of ones, and $\mathrm{diag}(\boldsymbol{w})$ is an $n_e \times n_e$ diagonal matrix with the elements of $\boldsymbol{w}$ down the diagonal. These are the specific expressions corresponding to (1) and (2) in Section 2.

The prior model induces the distributions of $M_F(\boldsymbol{x}_c) = \sum_{j=1}^{n_e} w_j Y(\boldsymbol{x}_c, \boldsymbol{x}_{e,j})$ for the mean $\mu_F(\cdot)$, and $V_F(\boldsymbol{x}_c) = \sum_{j=1}^{n_e} w_j (Y(\boldsymbol{x}_c, \boldsymbol{x}_{e,j}) - M_F(\boldsymbol{x}_c))^2$ for the variance $\sigma_F^2(\cdot)$. For the remainder of this paper, we suppress the dependence of $\sigma_F^2(\boldsymbol{x}_c)$ and $\mu_F(\boldsymbol{x}_c)$ on $F(\cdot)$, writing $\sigma^2(\boldsymbol{x}_c)$ as the measure of "flatness" and $\mu(\boldsymbol{x}_c)$ as the mean of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$ over $F(\cdot)$.

4. Sequential Algorithms

4.1. Overview

We propose sequential algorithms for determining the M-robust and V-robust control variables. Our algorithms are based on the concept of expected improvement (see Schonlau (1997) and Jones, Schonlau and Welch (1998)). Williams, Santner and Notz (2000) proposed a sequential expected improvement algorithm for choosing input sites at which to run the computer code when the goal is minimization of $\mu(\cdot)$. The algorithms presented here differ from those in Williams, Santner and Notz (2000) in three basic ways. First, we are not only interested in $\mu(\boldsymbol{x}_c)$, but also in $\sigma^2(\boldsymbol{x}_c) = \mathrm{Var}[y(\boldsymbol{x}_c, \boldsymbol{X}_e)]$, the variance of the distribution of $y(\boldsymbol{x}_c, \boldsymbol{X}_e)$. Second, we impose constraints on the feasible control variable values; when the goal is optimization of $\mu(\boldsymbol{x}_c)$, the feasible control variable region is defined by constraints on $\sigma^2(\boldsymbol{x}_c)$, and when the goal is optimization of $\sigma^2(\boldsymbol{x}_c)$, the feasible control variable region is defined by constraints on $\mu(\boldsymbol{x}_c)$. Finally, we propose a different method of selecting values of the environmental variables in Step 3 below. The steps of the algorithms reflect these differences.

Both algorithms are initialized by choosing an n-point initial design, denoted by $\boldsymbol{S}_n = \{\boldsymbol{x}_1^{tr}, \ldots, \boldsymbol{x}_n^{tr}\}$, at which to evaluate $y(\cdot)$. In our examples we

used ACED (Welch (1985)) to generate a maximin distance design within the set of Latin Hypercube designs, a space-filling design, although there are other possibilities such as Sobol´ or Niederreiter sequences (see Niederreiter (1992) and Owen (1995)). Let $\boldsymbol{Y}_n = [Y(\boldsymbol{x}_1^{tr}), \ldots, Y(\boldsymbol{x}_n^{tr})]^\top$ represent the vector of responses associated with the initial design sites in $\boldsymbol{S}_n$. In outline, the update steps of both sequential algorithms share the following elements.

1. Estimate the covariance parameter vector by $\hat{\boldsymbol{\gamma}}$, the mode of the posterior density of $\boldsymbol{\gamma}$ given $\boldsymbol{Y}_n$ from (15).
2. Choose the next control variable site $\boldsymbol{x}_c^*$ by an *improvement* criterion that depends on whether a M-robust or V-robust design is desired.
3. Choose the environmental variable site $\boldsymbol{x}_e^*$, corresponding to $\boldsymbol{x}_c^*$, to maximize the distance between $(\boldsymbol{x}_c^*, \boldsymbol{x}_e^*)$ and the point in $\boldsymbol{S}_n$ that is closest to $(\boldsymbol{x}_c^*, \boldsymbol{x}_e^*)$, i.e., so that
$$\boldsymbol{x}_e^* = \underset{\boldsymbol{x}_e \in \mathcal{X}_e}{\operatorname{argmax}}\, D[(\boldsymbol{x}_c^*, \boldsymbol{x}_e), \boldsymbol{S}_n],$$
where $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is a distance measure and $D(\boldsymbol{x}, \boldsymbol{S}_n) = \min\{d(\boldsymbol{x}, \boldsymbol{x}_{c,i}^{tr}) \ : \ 1 \leq i \leq n\}$.
4. Determine if the algorithm should be stopped. If not, set $\boldsymbol{S}_{n+1} = \boldsymbol{S}_n \bigcup \{(\boldsymbol{x}_c^*, \boldsymbol{x}_e^*)\}$, compute the response $y(\boldsymbol{x}_c^*, \boldsymbol{x}_e^*)$ and return to Step 1. If the stopping criterion is met, then the optimal robust setting for $\boldsymbol{x}_c$ is obtained using traditional optimization techniques with the posterior means of $M(\cdot)$ and $V(\cdot)$ substituted for $\mu(\cdot)$ and $\sigma^2(\cdot)$.

In Section 4.2 we present the criteria for adding points when the goal is to find the V-robust control variable values or the M-robust control variable values. For both criteria, the control variable portions of the input vectors in $\boldsymbol{S}_n$ will be denoted by $\boldsymbol{S}_n^C = \{\boldsymbol{x}_{c,1}^{tr}, \ldots, \boldsymbol{x}_{c,n}^{tr}\}$. For a given control variable value $\boldsymbol{x}_c$ we let $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c) = [Y(\boldsymbol{x}_c, \boldsymbol{x}_{e,1}), \ldots, Y(\boldsymbol{x}_c, \boldsymbol{x}_{e,n_e})]^\top$ and note that the random variables corresponding to (5) and (6) can be written as

$$M(\boldsymbol{x}_c) = \boldsymbol{w}^\top \boldsymbol{Y}_{n_e}(\boldsymbol{x}_c), \tag{7}$$

$$V(\boldsymbol{x}_c) = \boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)^\top \boldsymbol{A} \boldsymbol{Y}_{n_e}(\boldsymbol{x}_c), \tag{8}$$

where $\boldsymbol{A} = (I_{n_e} - \mathbf{1}_{n_e}\boldsymbol{w}^\top)\mathrm{diag}(\boldsymbol{w})(I_{n_e} - \mathbf{1}_{n_e}\boldsymbol{w}^\top)^\top$. Finally, let $\boldsymbol{M}_n = [M(\boldsymbol{x}_{c,1}^{tr}), \ldots, M(\boldsymbol{x}_{c,n}^{tr})]^\top$ be the vector of values for the mean response associated with $\boldsymbol{S}_n^C$.

4.2. Improvement criterion for finding robust designs

Formally, we define $\boldsymbol{x}_c^*$ to be V-robust if it satisfies $\boldsymbol{x}_c^* = \mathrm{argmin}_{\boldsymbol{x}_c \in \mathcal{X}_c}\, \sigma^2(\boldsymbol{x}_c)$ subject to either

$$\mu(\boldsymbol{x}_c^*) \leq \min_{\boldsymbol{x}_c \in \mathcal{X}_c}\ \mu(\boldsymbol{x}_c) + c, \qquad \text{or} \qquad \mu(\boldsymbol{x}_c^*) \leq c. \tag{9}$$

In words, the goal is to find $\boldsymbol{x}_c^*$ that minimizes $\sigma^2(\cdot)$ subject to $\mu(\boldsymbol{x}_c^*)$ being close to the global $\mu(\cdot)$ minimum $\min_{\boldsymbol{x}_c \in \mathcal{X}_c} \mu(\boldsymbol{x}_c)$, or satisfying a given constraint. The constraint equation (9) and the constraint parameter c are chosen to reflect the research objective. Use the left hand constraint with $c = 0$ to select an $\boldsymbol{x}_c$ that is a global minimizer of $\mu(\cdot)$. Use the right hand constraint if a set of possible mean values are to be considered, for example let $c = 5$ if any $\boldsymbol{x}_c$ with $\mu(\boldsymbol{x}_c) < 5$ is scientifically acceptable.

Alternatively, we define $\boldsymbol{x}_c^*$ to be M-robust if it satisfies $\boldsymbol{x}_c^* = \text{argmin}_{\boldsymbol{x}_c \in \mathcal{X}_c}$ $\mu(\boldsymbol{x}_c)$ subject to

$$\sigma^2(\boldsymbol{x}_c^*) \leq a \times \min_{\boldsymbol{x}_c \in \mathcal{X}_c} \sigma^2(\boldsymbol{x}_c) + c, \tag{10}$$

where $a \in \{0, [1, \infty)\}$, and $c \geq 0$. In words, the goal is to find $\boldsymbol{x}_c^*$ that minimizes $\mu(\cdot)$ subject to $\sigma^2(\boldsymbol{x}_c^*)$ being close to $\min_{\boldsymbol{x}_c \in \mathcal{X}_c} \sigma^2(\boldsymbol{x}_c)$ ($c \geq 0$ and $a \geq 1$) or satisfying a given constraint ($c > 0$ and $a = 0$).

To complete specification of the algorithms, we need to define the *improvement* criterion for selecting the next control variable value (Step 2). For the V-robust algorithm, we choose the next control variable $\boldsymbol{x}_c^*$ to maximize the expected improvement

$$I(\boldsymbol{x}_c) = E\{\max\{0, v_{min,f} - V(\boldsymbol{x}_c)\} \mid \boldsymbol{Y}_n, \boldsymbol{\gamma}\} \times P\{\text{ constraint } | \boldsymbol{Y}_n, \boldsymbol{\gamma}\}, \tag{11}$$

where the constraint is one of

$$M(\boldsymbol{x}_c) \ \leq \ M_{n,min} + c, \tag{12}$$

$$M(\boldsymbol{x}_c) \ \leq \ c. \tag{13}$$

The random variable $M_{n,min}$ is the minimum of $\boldsymbol{M}_n$, the vector of $M(\cdot)$ values at control variable sites in $\boldsymbol{S}_n^C$, and the constant $v_{min,f}$ is the current best guess at the constrained minimum of $\sigma^2(\cdot)$; $v_{min,f}$ is the *minimum* of the posterior expectations of $V(\cdot)$ for control variable values in $\boldsymbol{S}_n^C$ that appear to be in the *feasible* region. Formally, we let $M_{n,min} = \min\{M(\boldsymbol{x}_{c,i}^{tr}) \ : \ 1 \leq i \leq n\}$, and define the constant $v_{min,f}$ to be the minimum of $E[V(\boldsymbol{x}_c)|\boldsymbol{Y}_n, \boldsymbol{\gamma}]$ for $\boldsymbol{x}_c \in \boldsymbol{C}_n$, where $\boldsymbol{C}_n = \{\boldsymbol{x}_{c,i}^{tr} \in \boldsymbol{S}_n^C \ : \ M_{.025}(\boldsymbol{x}_{c,i}^{tr}) \leq \text{constraint}\}$ and $M_{.025}(\boldsymbol{x}_{c,i}^{tr})$ is the lower 2.5^{th} percentile of the posterior distribution of $M(\boldsymbol{x}_{c,i}^{tr})$ given $\boldsymbol{Y}_n$ and $\boldsymbol{\gamma}$ (see the Appendix). Thus, we choose $\boldsymbol{x}_c^*$ such that $\boldsymbol{x}_c^* = \text{argmax}_{\boldsymbol{x}_c \in \mathcal{X}_c} I(\boldsymbol{x}_c)$.

Alternatively, for the M-robust algorithm, we choose the next control variable site $\boldsymbol{x}_c^*$ to maximize

$$I(\boldsymbol{x}_c) = E[\max\{0, M_{min,f} - M(\boldsymbol{x}_c)\} \mid \boldsymbol{Y}_n, \boldsymbol{\gamma}] \times P[\text{ constraint } | \boldsymbol{Y}_n, \boldsymbol{\gamma}], \tag{14}$$

where the constraint is $V(\boldsymbol{x}_c) \ \leq \ a \times v_{n,min} + c$. The constant $v_{n,min}$ is the minimum of the posterior expected values of $V(\cdot)$ for control variable values

in $\boldsymbol{S}_n^C$, and the random variable $M_{min,f}$ is the *minimum* of $M(\cdot)$ at control variable values in $\boldsymbol{S}_n^c$ that appear to be in the *feasible* region. Formally, we let $v_{n,min} = \min\{E[V(\boldsymbol{x}_{c,i}^{tr})|\boldsymbol{Y}_n, \boldsymbol{\gamma}] : 1 \leq i \leq n\}$, and $M_{min,f} = \min\{M(\boldsymbol{x}_{c,i}^{tr}) : E[V(\boldsymbol{x}_{c,i}^{tr})|\boldsymbol{Y_n}, \boldsymbol{\gamma}] \leq a \times v_{n,min} + c\}$. Thus, we choose $\boldsymbol{x}_c^*$ such that $\boldsymbol{x}_c^* = \text{argmax}_{\boldsymbol{x}_c \in \mathcal{X}_c} I(\boldsymbol{x}_c)$.

The intuition behind these criteria is as follows. We choose the next control variable value site $\boldsymbol{x}_c^*$ to maximize the improvement in the objective function (either $V(\cdot)$ or $M(\cdot)$) over the current minimum of the posterior expected values of the objective function ($V(\cdot)$ or $M(\cdot)$) for control variable sites in the current design that *potentially* satisfy the constraint. We multiply this improvement by the probability that the constraint is satisfied so as not to "waste" observations in the infeasible region of the control variable space. The calculations necessary to determine (11) and (14) are outlined in the Appendix.

4.3. Posterior of $\boldsymbol{\gamma}$ given $\boldsymbol{Y}_n$

In the calculations of Steps 2 and 3 of both algorithms all distributions are given up to the unknown correlation parameter (vector) $\boldsymbol{\gamma}$. We adopt an empirical Bayes strategy and proceed with the algorithm by setting $\boldsymbol{\gamma}$ equal to its posterior mode (Step 1). The posterior density function of $\boldsymbol{\gamma}$ given $\boldsymbol{Y}_n$ satisfies

$$p(\boldsymbol{\gamma}|\boldsymbol{Y}_n) \propto p(\boldsymbol{\gamma}) \ |\boldsymbol{R}_{33}|^{-1/2} \ \left|\boldsymbol{F}_n^\top \boldsymbol{R}_{33}^{-1} \boldsymbol{F}_n\right|^{-1} \ [\hat{\tau}^2]^{(n-k)/2}, \tag{15}$$

where $p(\boldsymbol{\gamma})$ is the prior distribution on the permissible range of values for $\boldsymbol{\gamma}$ (see Handcock and Stein (1993)). The quantities $\boldsymbol{F}_n$ and $\boldsymbol{R}_{33}$ are the regression and correlation matrices associated with the vector $\boldsymbol{Y}_n$ (see the Appendix for more explanation), and $\hat{\tau}^2$ is the posterior estimate of τ^2 given $\boldsymbol{Y}_n$ and $\boldsymbol{\gamma}$ (see (20)). In Step 1 of the algorithms, we maximize (15) with $p(\boldsymbol{\gamma}) \propto 1$ although other choices are possible.

5. Examples

The following examples illustrate the performance of the M-robust and V-robust sequential algorithms. All calculations are performed using Model (4) for $Y(\boldsymbol{x}_c, \boldsymbol{x}_e)$ with constant mean $\boldsymbol{f}^\top(\boldsymbol{x})\boldsymbol{\beta} = \beta_0$, and product power exponential correlation function for $Z(\cdot)$ given by

$$R(\boldsymbol{h}) = \prod_{i=1}^{p} \exp\left(-\theta_i |h_i|^{\alpha_i}\right)$$

with individual scale and power parameters for each component of $(\boldsymbol{x}_c, \boldsymbol{x}_e)$. Here $\theta_i > 0$ and $0 < \alpha_i \leq 2$, so that $\boldsymbol{\gamma} = (\theta_1, \ldots, \theta_p, \alpha_1, \ldots, \alpha_p)$ is the unknown correlation function parameter.

5.1. V-robust 2-D example

We illustrate the algorithm for finding V-robust designs with a simple example. Consider the hypothetical $y(\cdot)$ shown in Figure 1 that depends on a real control variable and a real environmental variable, each on $(0, 1)$. We assume that $\boldsymbol{X}_e$ has a discretized uniform distribution on the 20 points $\{0.025, 0.075, \ldots, 0.975, 1\}$. For this example we wish to minimize $\sigma^2(x_c)$ subject to an absolute bound $\mu(x_c) \leq -0.08$.

The first step of the sequential algorithm involves choosing an initial set of design points at which to observe $y(\cdot)$. We use a space filling design, and, following the recommendations of Jones, Schonlau and Welch (1998), use 10 observations per input dimension. Figure 3 displays the 20-point maximin distance Latin hypercube design (+'s) used as the starting design for this example. The output $y(\cdot)$ is evaluated at each of the 20 points in the starting design, and the posterior mode of $\boldsymbol{\gamma}$ is obtained using (15). Figure 2 displays the true $\mu(x_c)$ (left panel) and $\sigma^2(x_c)$ (right panel), along with the posterior means of $M(x_c)$ and $V(x_c)$ given the data from the starting design and the posterior mode of $\boldsymbol{\gamma}$. Note that the constraint $\mu(x_c) \leq -0.08$ (dotted horizontal line in the left panel) restricts x_c to values in the approximate interval (0.176, 0.30). For this constraint the optimal setting of x_c is on the lower boundary of the feasible region at $x_c = 0.176$. If no points are added to the design and the posterior means of $M(x_c)$ and $V(x_c)$ based on the initial design are optimized we choose $x_c = 0.811$, a value of x_c that fails to satisfy the constraint of interest but appears to do so based on the 20-point predictions.

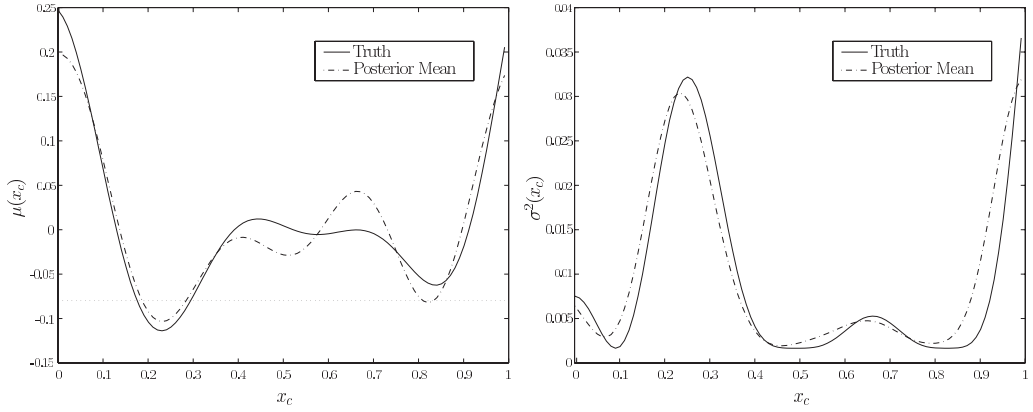

Figure 2. True $\mu(\cdot)$ (left panel) and $\sigma^2(\cdot)$ (right panel) and their posterior mean predictors based on the 20-point starting design for the V-robust example.

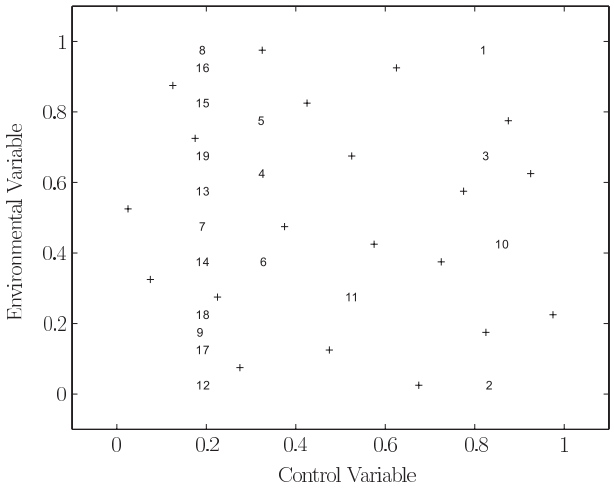

Figure 3. Locations of 39 points for the final design. +'s denote the 20 points in the initial design and the numbered sites are the sites added by the V-robust sequential algorithm in that order.

Using the V-robust sequential algorithm, points are added to the starting design until we reach a predefined stopping criterion. In general, we stop the algorithm when there is minimal improvement in adding points and/or when prediction in the feasible region is "accurate". For this example, we choose to stop the algorithm when a moving average of the improvement criterion is "small." The definition of "small" is problem specific since it is relative to the values of $\sigma^2(\cdot)$. One means of defining a small improvement is to require that the improvement be a small fraction (e.g., a thousandth) of the range for the posterior expected value of $V(\cdot)$ at the current stage. Here, we stop the algorithm when a 5-point moving average of the improvement is less than 0.00001. Figure 3 displays the 19 points that the algorithm added, and Figure 4 displays the final posterior means of $M(\cdot)$ and $V(\cdot)$ given the combined 39 (20 initial plus 19) point set of training data. Note the behavior of the algorithm. It appears to quickly recognize the region of the optimal x_c ($x_c \approx 0.2$), and then begins to sample a range of x_e values for that x_c. This is the desired behavior and, upon termination of the algorithm, the V-robust optimal value based on the posterior means matches the true V-robust optimal value of $x_c = 0.176$ to three decimal places.

5.2. M-robust 4-D example

We illustrate the M-robust sequential algorithm using the Branin function of Dixon and Szego (1978). The Branin function is defined on $\mathcal{X} = [-5, 10] \times [0, 15]$ by

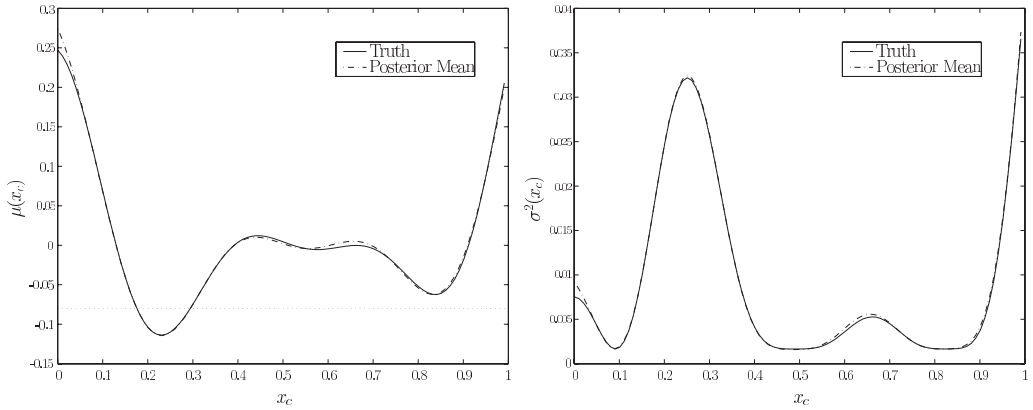

Figure 4. True $\mu(\cdot)$ and $\sigma^2(\cdot)$ and their posterior mean predictors based on the 39-point final design (20 points in initial design and 19 points added by the V-robust sequential algorithm).

$$z(x_1, x_2) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10.$$

The true response function has four inputs and is defined to be

$$y(x_1, x_2, x_3, x_4) = \frac{1}{30}z(x_1, x_2)z(x_3, x_4) + (x_1 - \pi)^2$$

with x_1 and x_2 being the control variables, $\boldsymbol{x}_c = (x_1, x_2)$, and x_3 and x_4 being the environmental variables, $\boldsymbol{x}_e = (x_3, x_4)$. Table 1 lists the assumed joint distribution of the environmental variables and Figure 5 displays the true $\mu(\boldsymbol{x}_c)$ (left panel) and the true $\sigma^2(\boldsymbol{x}_c)$ (right panel). We search for the M-robust control

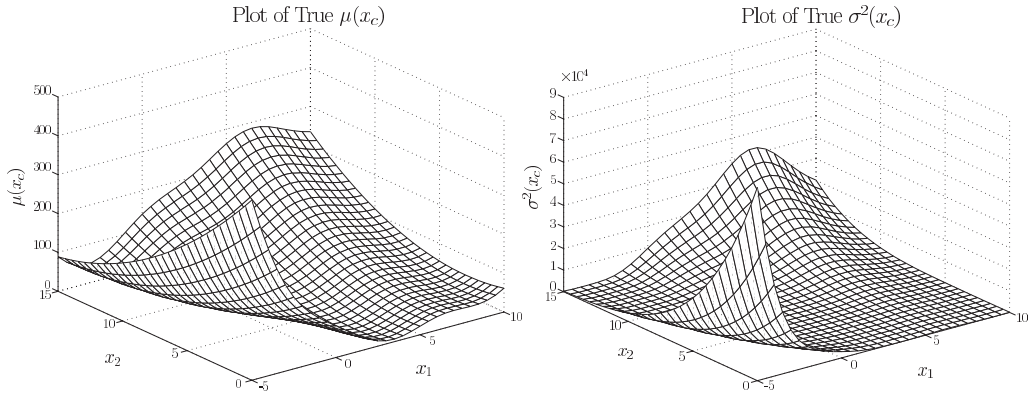

Figure 5. Plot of true $\mu(\boldsymbol{x}_c)$ (left panel) and true $\sigma^2(\boldsymbol{x}_c)$ (right panel) for 4-d example.

Table 1. Joint distribution of environmental variables in Section 5.2.

		x_3			
		-2	1	4	7
x_4	3.75	0.0375	0.0875	0.0875	0.0375
	7.5	0.0750	0.1750	0.1750	0.0750
	11.25	0.0375	0.0875	0.0875	0.0375

variable setting that minimizes $\mu(\boldsymbol{x}_c)$ subject to $\sigma^2(\boldsymbol{x}_c) < 10000$. The left panel of Figure 6 plots the $\boldsymbol{x}_c$ feasible region along with the global minimum of $\mu(\boldsymbol{x}_c)$, which occurs at the point $(\pi, 2.275)$ (denoted by $\triangle$ in the figure).

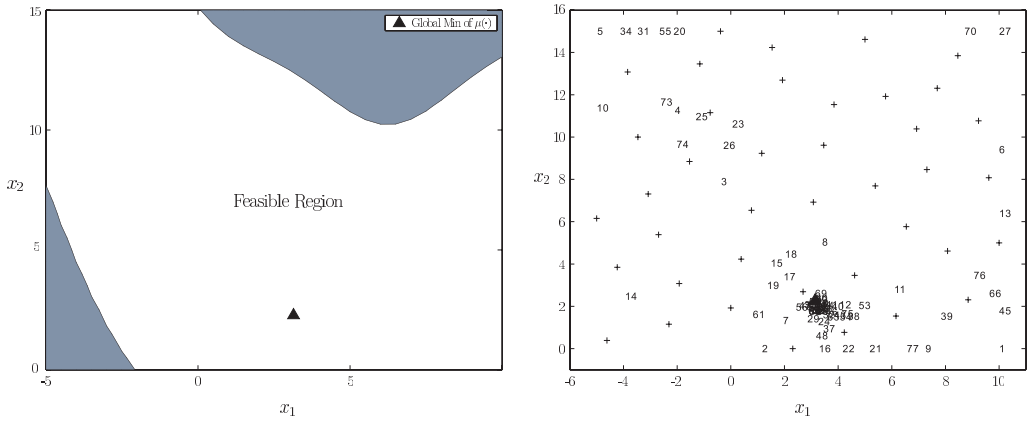

Figure 6. Plot of $\boldsymbol{x}_c$ feasible region (left panel) and 120 point final design projected into the control variable space (right panel). The +'s indicate initial design sites and the numbers indicate the additional design sites in the order they were chosen.

We begin the M-robust sequential algorithm by computing the response on a 40-point (again 10 observations for each dimension) maximin distance Latin hypercube design, obtaining the posterior mode of $\boldsymbol{\gamma}$ from (15), and plotting the posterior means of $M(\cdot)$ and $V(\cdot)$ based on the initial design in Figure 7. Using the M-robust criterion defined above, 80 points are added to the initial design. The right panel of Figure 6 plots the set of 120 (40 initial plus 80 added) final design sites projected onto the control variable space, and Figure 8 plots the posterior means of $M(\cdot)$ and $V(\cdot)$ based on the final 120 point design. Note the improvement in accuracy of the final predictors over the initial predictors, and note that the algorithm performs as desired by adding sites around the feasible region and slowly zeroing in on the true M-robust value of $(\pi, 2.275)$. Table 2 lists the value of the improvement and the predicted constrained minimizer as

points are added. After 80 points are added, the final predicted M-robust value is $(3.15, 2.25)$, a relative error of $0.32\% = 100\%\dot{(}3.15 - 3.14)/3.14$ for the first control variable and $1.1\% = 100\%\dot{(}2.25 - 2.275)/2.275$ for the second control variable in locating the true robust value of $\boldsymbol{x}_c$.

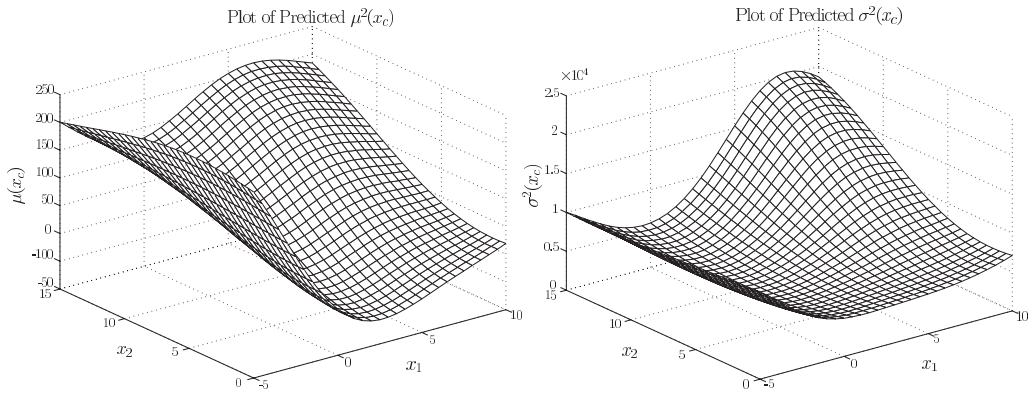

Figure 7. Posterior mean predictors of $\mu(\boldsymbol{x}_c)$ (left panel) and $\sigma^2(\boldsymbol{x}_c)$ (right panel) based on the initial 40 point design.

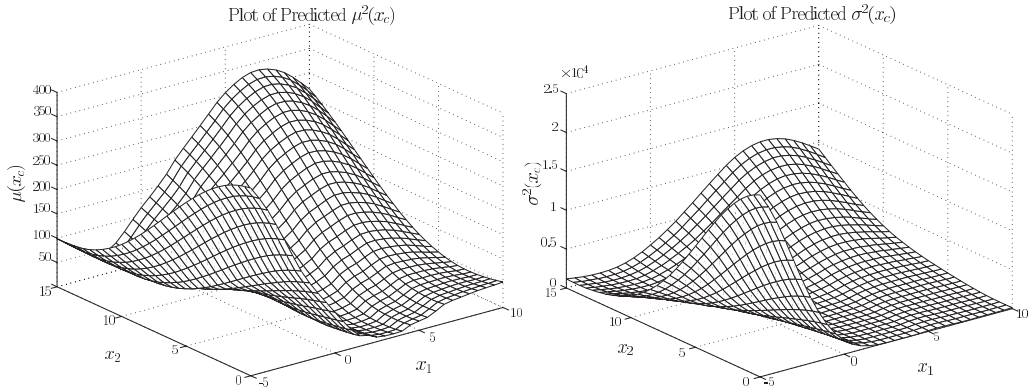

Figure 8. Posterior mean predictors of $\mu(\boldsymbol{x}_c)$ (left panel) and $\sigma^2(\boldsymbol{x}_c)$ (right panel) based on the final 120 point design.

Table 2. Summary results for the 4-D example in Section 5.2.

# Points Added	Improvement	Predicted Minimizer
1	2.904673	(2.65, 0.00)
40	0.236627	(3.31, 2.02)
60	0.020322	(2.92, 1.98)
75	0.024939	(3.15, 1.97)
80	0.019231	(3.15, 2.25)

6. Discussion

The numerical optimization in Steps 1 and 2 of both algorithms can be computationally challenging, especially as the dimension of the input space increases. In particular, for computational savings and for larger dimensional problems (i.e., > 10 dimensions for example) alternative choices may need to be made for this algorithm to be computationally feasible. For example, correlation functions with fewer parameters may be necessary, and/or updating of the correlation parameter estimates in Step 1 could be completed only after adding *groups* of points to the existing design and/or by using the previous correlation parameter estimates as a starting point for the current stage's numerical optimization. Additionally, in Step 2, the algorithm calls for numerical optimization of the improvement to obtain the value of the next control portion of the input at which to observe $y(\cdot)$. Promising starting values for this optimization can be obtained by evaluating the improvement criterion on a grid of points in $\mathcal{X}_c$.

Step 3 of the algorithm presents a heuristic scheme for selection of the value of the next environmental portion of the input at which to observe $y(\cdot)$. Space-filling designs, such as Latin hypercubes (McKay, Beckman and Conover (1979)) or distance-based designs, are popular single stage designs in the computer experiments literature. The intuition behind these designs is that observations should be spread out in order to "cover" the input space. For this goal, space-filling designs seem a natural choice, and for this reason, we suggest selection of $\boldsymbol{x}_e$ via a distance-based criterion that also attempts to "spread" out observations by selecting $\boldsymbol{x}_e$ so that the selected point $(\boldsymbol{x}_c^*, \boldsymbol{x}_e)$ is not too "close" to points that have already been observed. This criterion is simple to understand and to implement; however, many other methods of choosing $\boldsymbol{x}_e$ are reasonable and may prove effective (see for example Williams et al. (2000)).

As seen in the examples, the stopping criterion for the algorithm is problem specific. Generally, we want to stop the algorithm when our predictions are *accurate* and/or the improvement is small. When optimizing, we suggest stopping the algorithm when a moving average of the improvement is "small". A moving average is used because the improvement need not be a strictly decreasing function (the use of updated correlation parameter estimates and additional information from the newly observed response can cause larger improvements in the current cycle of the algorithm than those previously observed). The definition of "small" improvement is problem specific since it depends on the scale of the objective function. For the V-robust algorithm a "small" improvement may be determined as a small fraction of the range of the posterior expected values of $V(\cdot)$, and for the M-robust algorithm a "small" improvement may be a small fraction of the range of the posterior expected values of $M(\cdot)$.

Both the V-robust and M-robust sequential algorithms require a starting design at which the responses are calculated. In the examples above, maximin distance LHS designs were used as starting designs. However, any other *space-filling* design may be appropriate for the goals presented here. Additionally, allocation of runs to the initial design and subsequent sequential design is an important consideration. "Too many" observations in the intial design will "waste" observations, while too few may lead to poor correlation parameter estimates and a larger number of sequential steps to find the optimum. Generating space-filling designs and determining which are best suited to computer experiments is an area of active research. Another promising area for future research is in developing formal results for the choice of sample size in computer experiments. In the examples, we use the Jones, Schonlau and Welch (1998) suggestion of 10 observations per input dimension, which we have found to be a reasonable rule of thumb.

In this paper we have considered the case of a single deterministic response $y(\cdot)$. Straightforward extensions of these concepts can be made for the case where the response contains additive measurement error. Many computer experiments involve multiple, related responses; for such situations, extensions of the above algorithms and concepts may be appropriate following meaningful modeling of the related responses as in Williams et al. (2003).

Acknowledgement

We would like to thank Brian Williams for helpful conversation in developing these ideas and two referees and an associate editor for suggestions that improved the presentation.

Appendix

In the following we present the details of the calculations required in computing the expected improvement for the V-robust algorithm (11). Parallel calculations for the M-robust algorithm, (14), can be found in Lehman, Santner and Notz (2002). Let $N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote the p-variate normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Additionally, let $\mathcal{T}_q(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$ denote the q-variate T distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\nu\boldsymbol{\Sigma}/(\nu-2)$ if $\nu > 2$, i.e., the q-variate random variable with density

$$\frac{\Gamma[(\nu+q)/2]}{|\boldsymbol{\Sigma}|^{1/2}(\nu\Pi)^{q/2}\Gamma[\nu/2]}\left(1 \; + \; \frac{1}{\nu}(\boldsymbol{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)^{-(\nu+q)/2} \tag{16}$$

for $\boldsymbol{x} \in \mathbb{R}^q$. The following lemma, appearing in O'Hagan (1992), is used throughout the Appendix.

Lemma A.1. *Suppose $\boldsymbol{U}_i$ for $i \in \{1,2\}$ denote $q_i \times 1$ random vectors having the Gaussian distribution*

$$\begin{pmatrix} \boldsymbol{U}_1 \\ \boldsymbol{U}_2 \end{pmatrix} \mid \boldsymbol{\beta}, \sigma^2 \sim \mathcal{N}_{q_1+q_2}\left[\begin{pmatrix} \boldsymbol{F}_1 \\ \boldsymbol{F}_2 \end{pmatrix}\boldsymbol{\beta}, \sigma^2 \begin{pmatrix} \boldsymbol{R}_{11}\ \boldsymbol{R}_{12} \\ \boldsymbol{R}_{12}^\top\ \boldsymbol{R}_{22} \end{pmatrix}\right],$$

where $\boldsymbol{\beta} \in \mathbb{R}^k$ and $\sigma^2 > 0$. Assuming that each of the elements of $\boldsymbol{F}_i$ and $\boldsymbol{R}_{ij}$ are known, each $\boldsymbol{F}_i$ has full column rank, the correlation matrix is positive definite and the parameter vector $\boldsymbol{\beta}, \sigma^2$ has the noninformative prior $[\boldsymbol{\beta}, \sigma^2] \propto 1/\sigma^2$, the posterior distribution of $\boldsymbol{U}_1$ given $\boldsymbol{U}_2$ is $\mathcal{T}_{q_1}(\boldsymbol{m}_{1|2}, \widehat{\sigma^2}\boldsymbol{R}_{1|2}, q_2 - k)$ where $\boldsymbol{m}_{1|2} = \boldsymbol{F}_1\hat{\boldsymbol{\beta}} + \boldsymbol{R}_{12}\boldsymbol{R}_{22}^{-1}(\boldsymbol{U}_2 - \boldsymbol{F}_2\hat{\boldsymbol{\beta}})$, $\hat{\boldsymbol{\beta}} = (\boldsymbol{F}_2^\top \boldsymbol{R}_{22}^{-1}\boldsymbol{F}_2)^{-1}\boldsymbol{F}_2^\top \boldsymbol{R}_{22}^{-1}\boldsymbol{U}_2$, $\widehat{\sigma^2} = [\boldsymbol{U}_2^\top \boldsymbol{R}_{22}^{-1}\boldsymbol{U}_2 - \hat{\boldsymbol{\beta}}^\top(\boldsymbol{F}_2^\top \boldsymbol{R}_{22}^{-1}\boldsymbol{F}_2)\hat{\boldsymbol{\beta}}]/(q_2 - k)$ and $\boldsymbol{R}_{1|2} = \boldsymbol{R}_{11}$ - $\boldsymbol{R}_{12}\boldsymbol{R}_{22}^{-1}\boldsymbol{R}_{12}^\top$ + $(\boldsymbol{F}_1 - \boldsymbol{R}_{12}\boldsymbol{R}_{22}^{-1}\boldsymbol{F}_2)$ $(\boldsymbol{F}_2^\top \boldsymbol{R}_{22}^{-1}\boldsymbol{F}_2)^{-1}$ $(\boldsymbol{F}_1 - \boldsymbol{R}_{12}\boldsymbol{R}_{22}^{-1}\boldsymbol{F}_2)^\top$.

We begin calculation of (11) with the joint distribution of $(\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$, $\boldsymbol{Y}_{n,n_e}$, $\boldsymbol{Y}_n)$ given $\boldsymbol{\beta}$, τ^2 and $\boldsymbol{\gamma}$, where $\boldsymbol{Y}_{n,n_e} = (Y(\boldsymbol{x}_{c,1}^{tr}, \boldsymbol{x}_{e,1}), \ldots, Y(\boldsymbol{x}_{c,1}^{tr}, \boldsymbol{x}_{e,n_e}), \ldots,$ $Y(\boldsymbol{x}_{c,n}^{tr}, \boldsymbol{x}_{e,1}), \ldots, Y(\boldsymbol{x}_{c,n}^{tr}, \boldsymbol{x}_{e,n_e}))^\top$ is the $(n \times n_e) \times 1$ vector of responses at control sites in $\boldsymbol{S}_n^C$ paired with each support point for the environmental variables. From (4) the joint distribution of $[\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$, $\boldsymbol{Y}_{n,n_e}$, $\boldsymbol{Y}_n]$ given $(\boldsymbol{\beta}, \tau^2, \boldsymbol{\gamma})$, is multivariate normal with mean $(\boldsymbol{F}_{n_e}^\top(\boldsymbol{x}_c)$, $\boldsymbol{F}_{n,n_e}^\top$, $\boldsymbol{F}_n^\top)^\top\boldsymbol{\beta}$ and variance-covariance matrix $\tau^2((\boldsymbol{\Sigma}_{pq}))$ for $p, q \in \{1,2,3\}$, where the components are defined next. The vectors $\boldsymbol{F}_{n_e}(\boldsymbol{x}_c) = [\boldsymbol{f}(\boldsymbol{x}_c, \boldsymbol{x}_{e,1}), \ldots, \boldsymbol{f}(\boldsymbol{x}_c, \boldsymbol{x}_{e,n_e})]^\top$, $\boldsymbol{F}_{n,n_e} = [\boldsymbol{f}(\boldsymbol{x}_{c,1}^{tr}, \boldsymbol{x}_{e,1}), \ldots,$ $\boldsymbol{f}(\boldsymbol{x}_{c,1}^{tr}, \boldsymbol{x}_{e,n_e}), \ldots, \boldsymbol{f}(\boldsymbol{x}_{c,n}^{tr}, \boldsymbol{x}_{e,n_e})]^\top$ and $\boldsymbol{F}_n = [\boldsymbol{f}(\boldsymbol{x}_1^{tr}), \ldots, \boldsymbol{f}(\boldsymbol{x}_n^{tr})]^\top$ are the regression matrices for $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$, $\boldsymbol{Y}_{n,n_e}$ and $\boldsymbol{Y}_n$ respectively. The indices $p, q \in \{1,2,3\}$ for the covariance matrices $\boldsymbol{\Sigma}_{pq}$ correspond to the three components $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$, $\boldsymbol{Y}_{n,n_e}$ and $\boldsymbol{Y}_n$ in this order, so that, for example, $\mathrm{Cov}[\boldsymbol{Y}_{n,n_e}, \boldsymbol{Y}_n] = \tau^2\boldsymbol{\Sigma}_{23}$.

Because Gaussian random vectors remain Gaussian under linear transformations we have

$$\begin{pmatrix} \boldsymbol{Y}_{n_e}(\boldsymbol{x}_c) \\ \boldsymbol{M}_n \\ \boldsymbol{Y}_n \end{pmatrix} | \boldsymbol{\beta}, \tau^2, \boldsymbol{\gamma} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{F}_{n_e}(\boldsymbol{x}_c) \\ \bar{\boldsymbol{F}}_{n,n_e} \\ \boldsymbol{F}_n \end{pmatrix}\boldsymbol{\beta}, \tau^2 \begin{pmatrix} \boldsymbol{R}_{11}\ \boldsymbol{R}_{12}\ \boldsymbol{R}_{13} \\ \cdot\ \ \boldsymbol{R}_{22}\ \boldsymbol{R}_{23} \\ \cdot\ \ \cdot\ \ \boldsymbol{R}_{33} \end{pmatrix}\right), \tag{17}$$

where $\bar{\boldsymbol{F}}_{n,n_e} = (I_n \otimes \boldsymbol{w}^\top)\boldsymbol{F}_{n,n_e}$, $\boldsymbol{R}_{11} = \boldsymbol{\Sigma}_{11}$, $\boldsymbol{R}_{13} = \boldsymbol{\Sigma}_{13}$, $\boldsymbol{R}_{33} = \boldsymbol{\Sigma}_{33}$, $\boldsymbol{R}_{12}$=$\boldsymbol{\Sigma}_{12}(I_n$ $\otimes \boldsymbol{w}^\top)^\top$, $\boldsymbol{R}_{23}$=$(I_n \otimes \boldsymbol{w}^\top)\boldsymbol{\Sigma}_{23}$ and $\boldsymbol{R}_{22}$=$(I_n \otimes \boldsymbol{w}^\top)\boldsymbol{\Sigma}_{22}(I_n \otimes \boldsymbol{w}^\top)^\top$.

To calculate $E[\max\{0, v_{min,f} - V(\boldsymbol{x}_c)\}|\boldsymbol{Y}_n, \boldsymbol{\gamma}]$ we first compute the constant $v_{min,f}$. From (8), $V(\boldsymbol{x}_c) = \boldsymbol{Y}_{n_e}^\top(\boldsymbol{x}_c)\boldsymbol{A}\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$ is a quadratic form in $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$ for all $\boldsymbol{x}_c$. Using Lemma A.1, the posterior distribution of $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$ given $\boldsymbol{Y}_n$ and $\boldsymbol{\gamma}$ is

$$[\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)|\boldsymbol{Y}_n, \boldsymbol{\gamma}] \ \sim\ \mathcal{T}_{n_e}(\boldsymbol{m}, \hat{\tau}^2\boldsymbol{R}, n - k), \tag{18}$$

where $\boldsymbol{m} = \boldsymbol{F}_{n_e}(\boldsymbol{x}_c)\hat{\boldsymbol{\beta}} + \boldsymbol{R}_{13}\boldsymbol{R}_{33}^{-1}[\boldsymbol{Y}_n - \boldsymbol{F}_n\hat{\boldsymbol{\beta}}]$, $\boldsymbol{R} = \boldsymbol{R}_{11} - \boldsymbol{R}_{13}\boldsymbol{R}_{33}^{-1}\boldsymbol{R}_{13}^{\top} + (\boldsymbol{F}_{n_e}(\boldsymbol{x}_c) - \boldsymbol{R}_{13}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)(\boldsymbol{F}_n^{\top}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)^{-1}\ (\boldsymbol{F}_{n_e}(\boldsymbol{x}_c) - \boldsymbol{R}_{13}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)^{\top}$, and

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{F}_n^{\top}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)^{-1}\boldsymbol{F}_n^{\top}\boldsymbol{R}_{33}^{-1}\boldsymbol{Y}_n, \tag{19}$$

$$\hat{\tau}^2 = \frac{\boldsymbol{Y}_n^{\top}\boldsymbol{R}_{33}^{-1}\boldsymbol{Y}_n \ - \ \hat{\boldsymbol{\beta}}^{\top}(\boldsymbol{F}_n^{\top}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)\hat{\boldsymbol{\beta}}}{n-k}. \tag{20}$$

Applying the well known formula for expectations of a quadratic form we obtain

$$\mathrm{E}[V(\boldsymbol{x}_c)|\boldsymbol{Y}_n, \boldsymbol{\gamma}] = \frac{n-k}{n-k-2}\mathrm{trace}[\hat{\tau}^2\boldsymbol{R}\boldsymbol{A}] + \boldsymbol{m}^{\top}\boldsymbol{A}\boldsymbol{m}. \tag{21}$$

We calculate (21) for each control variable value in $\boldsymbol{C}_n$, set $v_{min,f}$ as the minimum of these expected values, and compute $E[\max\{0, v_{min,f} - V(\boldsymbol{x}_c)\}|\boldsymbol{Y}_n, \boldsymbol{\gamma}]$ via Monte Carlo by generating N_v samples of $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$ from the distribution of $\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c)$ given $\boldsymbol{Y}_n$ and $\boldsymbol{\gamma}$.

A formula for the posterior probability of constraint (12) is obtained via iterated expectations and Monte Carlo. We have

$$P[M(\boldsymbol{x}_c) \le M_{n,min} + c|\boldsymbol{Y}_n, \boldsymbol{\gamma}] = E_{\boldsymbol{M}_n|\boldsymbol{Y}_n,\boldsymbol{\gamma}}\left[P(M(\boldsymbol{x}_c) \le M_{n,min} + c \mid \boldsymbol{M}_n, \boldsymbol{Y}_n, \boldsymbol{\gamma})\right]. \tag{22}$$

To compute the inner expectation we apply a linear transformation and Lemma A.1 to (17) and obtain

$$[M(\boldsymbol{x}_c)|\boldsymbol{Z}_{2n}, \boldsymbol{\gamma}] \sim \mathcal{T}_1(\mu_{M,1}, \tilde{\tau}^2\sigma_{M,1}, 2n-k), \tag{23}$$

where $\boldsymbol{Z}_{2n} = (\boldsymbol{M}_n^{\top}, \boldsymbol{Y}_n^{\top})^{\top}$. Letting $\boldsymbol{R}_{1,23} = (\boldsymbol{R}_{12}, \boldsymbol{R}_{13})$, $\boldsymbol{F}_{2n} = \left(\bar{\boldsymbol{F}}_{n,n_e}^{\top}, \boldsymbol{F}_n^{\top}\right)^{\top}$ and $\boldsymbol{R}_{2n} = \{\boldsymbol{R}_{ij}\}$ for $i, j \in \{2, 3\}$ (the $2n \times 2n$ lower block of the correlation matrix in (17)), we have $\tilde{\tau}^2 = \boldsymbol{Z}_{2n}^{\top}\boldsymbol{R}_{2n}^{-1}\boldsymbol{Z}_{2n} - \widetilde{\boldsymbol{\beta}}^{\top}(\boldsymbol{F}_{2n}^{\top}\boldsymbol{R}_{2n}^{-1}\boldsymbol{F}_{2n})\widetilde{\boldsymbol{\beta}}/(2n-k)$, $\widetilde{\boldsymbol{\beta}} = (\boldsymbol{F}_{2n}^{\top}\boldsymbol{R}_{2n}^{-1}\boldsymbol{F}_{2n})^{-1}\boldsymbol{F}_{2n}^{\top}\boldsymbol{R}_{2n}^{-1}\boldsymbol{Z}_{2n}$, and

$$\mu_{M,1} = \boldsymbol{w}^{\top}\boldsymbol{F}_{n_e}(\boldsymbol{x}_c)\widetilde{\boldsymbol{\beta}} + \boldsymbol{w}^{\top}\boldsymbol{R}_{1,23}(\boldsymbol{Z}_{2n} - \boldsymbol{F}_{2n}\widetilde{\boldsymbol{\beta}}), \tag{24}$$

$$\begin{aligned}\sigma_{M,1} = {} & \boldsymbol{w}^{\top}\boldsymbol{R}_{11}\boldsymbol{w} - \boldsymbol{w}^{\top}\boldsymbol{R}_{1,23}\boldsymbol{R}_{2n}^{-1}\boldsymbol{R}_{1,23}^{\top}\boldsymbol{w} \\ & + \boldsymbol{w}^{\top}(\boldsymbol{F}_{n_e}(\boldsymbol{x}_c) - \boldsymbol{R}_{1,23}\boldsymbol{R}_{2n}^{-1}\boldsymbol{F}_{2n})(\boldsymbol{F}_{2n}^{\top}\boldsymbol{R}_{2n}^{-1}\boldsymbol{F}_{2n})^{-1}(\boldsymbol{F}_{n_e}(\boldsymbol{x}_c) \\ & - \boldsymbol{R}_{1,23}\boldsymbol{R}_{2n}^{-1}\boldsymbol{F}_{2n})^{\top}\boldsymbol{w}.\end{aligned} \tag{25}$$

Thus, the inner expectation is

$$P[M(\boldsymbol{x}_c) \le M_{n,min} + c \mid \boldsymbol{M}_n, \boldsymbol{Y}_n, \boldsymbol{\gamma}] = \mathrm{T}_{2n-k}\left[(M_{n,min} + c - \mu_{M,1})/\sqrt{\tilde{\tau}^2\sigma_{M,1}}\right], \tag{26}$$

where $T_{\nu}(\cdot)$ is the univariate T cdf with ν degrees of freedom. The outer expectation is obtained via Monte Carlo. We generate N_{μ} random samples from the

distribution of $[\boldsymbol{M}_n|\boldsymbol{Y}_n, \boldsymbol{\gamma}]$, and compute (26) for each sample. The value for (22) is then obtained as the average of these N_μ quantities.

The distribution of $[\boldsymbol{M}_n|\boldsymbol{Y}_n, \boldsymbol{\gamma}]$ is computed by applying Lemma A.1 to the $[\boldsymbol{M}_n, \boldsymbol{Y}_n]$ portion of distribution (17). We have $[\boldsymbol{M}_n|\boldsymbol{Y}_n, \boldsymbol{\gamma}] \sim \mathcal{T}_n(\boldsymbol{\mu}_{M_n}, \hat{\tau}^2\boldsymbol{\Sigma}_{M_n}, n-k)$, where $\boldsymbol{\mu}_{M_n} = \bar{\boldsymbol{F}}_{n,n_e}\hat{\boldsymbol{\beta}} + \boldsymbol{R}_{23}\boldsymbol{R}_{33}^{-1}(\boldsymbol{Y}_n - \boldsymbol{F}_n\hat{\boldsymbol{\beta}})$, $\hat{\boldsymbol{\beta}}$ as in (19), $\hat{\tau}^2$ as in (20), and $\boldsymbol{\Sigma}_{M_n} = \boldsymbol{R}_{22} - \boldsymbol{R}_{23}\boldsymbol{R}_{33}^{-1}\boldsymbol{R}_{23}^\top + (\bar{\boldsymbol{F}}_{n,n_e} - \boldsymbol{R}_{23}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)(\boldsymbol{F}_n^\top\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)^{-1}(\bar{\boldsymbol{F}}_{n,n_e} - \boldsymbol{R}_{23}\boldsymbol{R}_{33}^{-1}\boldsymbol{F}_n)^\top$.

A formula for the constraint $P[M(\boldsymbol{x}_c) \le c \mid \boldsymbol{Y}_n, \boldsymbol{\gamma}]$, the posterior probability of (13) follows in a similar fashion. Applying a linear transformation to the distribution of $[\boldsymbol{Y}_{n_e}(\boldsymbol{x}_c), \boldsymbol{Y}_n]$ given $(\boldsymbol{\beta}, \tau^2, \boldsymbol{\gamma})$, and then applying Lemma A.1 to the result gives $[M(\boldsymbol{x}_c)|\boldsymbol{Y}_n, \boldsymbol{\gamma}] \sim \mathcal{T}_1(\mu_{M,2}, \hat{\tau}^2\sigma_{M,2}, n-k)$, where the formulas for $\mu_{M,2}$ and $\sigma_{M,2}$ are identical to formulas (24) and (25) for $\mu_{M,1}$ and $\sigma_{M,1}$ with $\tilde{\boldsymbol{\beta}}$ replaced by $\hat{\boldsymbol{\beta}}$, $\boldsymbol{R}_{1,23}$ replaced by $\boldsymbol{R}_{13}$, $\boldsymbol{Z}_{2n}$ replaced by $\boldsymbol{Y}_n$, $\boldsymbol{F}_{2n}$ replaced by $\boldsymbol{F}_n$, and $\boldsymbol{R}_{2n}$ replaced by $\boldsymbol{R}_{33}$. We obtain $P[M(\boldsymbol{x}_c) \le c \mid \boldsymbol{Y}_n, \boldsymbol{\gamma}] = \mathrm{T}_{n-k}((c-\mu_{M,2})/\sqrt{\hat{\tau}^2\sigma_{M,2}})$, where $T_\nu(\cdot)$ is the univariate T cdf with ν degrees of freedom. Analagous calculations for the M-robust improvement (14) can be found in Lehman et al. (2002).

References

Chang, P. B. and Williams, B. J., Notz, W. I., Santner, T. J. and Bartel, D. L. (1999). Robust optimization of total joint replacements incorporating environmental variables. *J. Biomechan. Engineering* **121**, 304-310.

Dixon, L. C. W. and Szego, G. P. (1978). The global optimisation problem: an introduction, In *Towards Global Optimisation* **2** (Edited by L. C. W. Dixon and G. P. Szego), 1-15. North Holland, Amsterdam.

Handcock, M. S. and Stein, M. L. (1993). A Bayesian analysis of kriging. *Technometrics* **35**, 403-410.

Haylock, R. G. and O'Hagan, A. (1996). *On inference for outputs of computationally expensive algorithms with uncertainty on the inputs.* In *Bayesian Statistics* **5** (Edited by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith), 629-637. Oxford University Press.

Huber, P. J. (1981). *Robust Statistics.* Wiley, New York.

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

Koehler, J. R. and Owen, A. B. (1996). Computer experiments In *Handbook of Statistics* **13** (Edited by S. Ghosh and C. R. Rao), 261-308. Elsevier Science B.V.

Lehman, J. S., Santner, T. J. and Notz, W. I. (2002). Robust parameter design for computer experiments. Technical Report 708, Department of Statistics, The Ohio State University.

McKay, M. D., Beckman, R. J. and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, **21**, 239-245.

Niederreiter, H. (1992). *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia.

O'Hagan, A. (1992). Some Bayesian numerical analysis. In *Bayesian Statistics* **4** (Edited by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith), 345-363. Oxford University Press.

O'Hagan, A. and Haylock, R. G. (1997). Bayesian uncertainty analysis and radiological protection. In *Statistics for the Environment* **3** (Edited by V. Barnett and K. F. Turkman), 109-128. Wiley, New York.

O'Hagan, A., Kennedy, M. C. and Oakley, J. E. (1999). Uncertainty analysis and other inference tools for complex computer codes. In *Bayesian Statistics* **6** (Edited by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith), 503-524. Oxford University Press.

Sacks, J. and Schiller, S. B. and Welch, W. J. (1989). Designs for computer experiments, *Technometrics*, **31**, 41-47.

Schonlau, M. (1997). Computer experiments and global optimization. Ph.D. thesis. Department of Statistics and Actuarial Science, University of Waterloo.

Schonlau, M. and Welch, W. and Jones, D. (1998). Global versus local search in constrained optimization of computer models. In *New Developments and Applications in Experimental Design* **34** (Edited by N. Flournoy, W. F. Rosenberger and W. K. Wong), 11-25. Institute of Mathematical Statistics.

Welch, W. J. (1985). ACED: Algorithms for the construction of experimental designs. *Amer. Statist.* **39**, 146.

Welch, W. J., Yu, T.-K., Kang, S. M. and Sacks, J. (1990). Computer experiments for quality control by parameter design, *J. Quality Tech.* **22**, 15-22.

Williams, B. J., Lehman, J. S., Santner, T. J. and Notz, W. I. (2003). Constrained optimization of computer experiments having multiple responses. Submitted.

Williams, B. J., Santner, T. J. and Notz, W. I. (2000). Sequential design of computer experiments to minimize integrated response functions. *Statist. Sinica* **10**, 1133-1152.

Statistics and Data Analysis Systems, Battelle Memorial Institute, 505 King Avenue, Columbus, OH 43201, U.S.A.

E-mail: lehmanj2@battelle.org

Department of Statistics, The Ohio State University Cockins Hall, 1958 Neil Ave., Columbus, OH 43210, U.S.A.

E-mail: tjs@stat.ohio-state.edu

Department of Statistics, The Ohio State University Cockins Hall, 1958 Neil Ave., Columbus, OH 43210, U.S.A.

E-mail: win@stat.ohio-state.edu

(Received October 2002; accepted August 2003)