Bayesian Design and Analysis of Computer Experiments: Use of Derivatives in Surface Prediction

Max D. Morris; Toby J. Mitchell; Donald Ylvisaker

Technometrics, Vol. 35, No. 3. (Aug., 1993), pp. 243-255.

Stable URL:
http://links.jstor.org/sici?sici=0040-1706%28199308%2935%3A3%3C243%3ABDAAOC%3E2.0.CO%3B2-E

Technometrics is currently published by American Statistical Association.

Your use of the JSTOR archive indicates your acceptance of JSTOR's Terms and Conditions of Use, available at http://www.jstor.org/about/terms.html. JSTOR's Terms and Conditions of Use provides, in part, that unless you have obtained prior permission, you may not download an entire issue of a journal or multiple copies of articles, and you may use content in the JSTOR archive only for your personal, non-commercial use.

Please contact the publisher regarding any further use of this work. Publisher contact information may be obtained at http://www.jstor.org/journals/astata.html.

Each copy of any part of a JSTOR transmission must contain the same copyright notice that appears on the screen or printed page of such transmission.

JSTOR is an independent not-for-profit organization dedicated to creating and preserving a digital archive of scholarly journals. For more information regarding JSTOR, please contact support@jstor.org.

http://www.jstor.org/
Fri Jul 7 19:07:37 2006

In the Public Domain

TECHNOMETRICS, AUGUST 1993, VOL. 35, NO. 3

Bayesian Design and Analysis of Computer Experiments: Use of Derivatives in Surface Prediction

Max D. Morris and Toby J. Mitchell

Mathematical Sciences Section, EPM Division
Oak Ridge National Laboratory
Oak Ridge, TN 37831-6367

Donald Ylvisaker

Department of Mathematics
University of California at Los Angeles
Los Angeles, CA 90024

This article is concerned with the problem of predicting a deterministic response function y_0 over a multidimensional domain T, given values of y_0 and all of its first derivatives at a set of design sites (points) in T. The intended application is to computer experiments in which y_0 is an output from a computer model of a physical system and each point in T represents a particular configuration of the input parameters. It is assumed that the first derivatives are already available (e.g., from a sensitivity analysis) or can be produced by the code that implements the model. A Bayesian approach in which the random function that represents prior uncertainty about y_0 is taken to be a stationary Gaussian stochastic process is used. The calculations needed to update the prior given observations of y_0 and its first derivatives at the design sites are given and are illustrated in a small example. The issue of experimental design is also discussed, in particular the criterion of maximizing the reduction in entropy, which leads to a kind of D optimality. It is shown that, for certain classes of correlation functions in which the intersite correlations are very weak, D-optimal designs necessarily maximize the minimim distance between design sites. A simulated annealing algorithm is described for constructing such maximin distance designs. An example is given based on a demonstration model of eight inputs and one output, in which predictions based on a maximin design, a Latin hypercube design, and two compromise designs are evaluated and compared.

KEY WORDS: Bayesian prediction; Computer model; Interpolation; Latin hypercube design; Maximin distance design; Sensitivity analysis.

Computer codes that are based on mathematical models of physical or behavioral systems have become important tools in virtually all fields of scientific research. As a surrogate for a real system, such a *computer model* can be subjected to experimentation, the goal being to predict how that system would behave under certain conditions. In each experimental run, the code is used to generate a vector of *response variables* $y_0(t)$ from a vector of *design variables* $t = (t_1, t_2, \ldots, t_k)$. For convenience of exposition here, we shall consider t to be a subset of the inputs to the code and y_0 a single (scalar) output. More generally, t is a set of variables that determines the inputs and y_0 is a set of variables that is computed from the outputs. The function y_0 implicitly defined in this way over some domain T is deterministic; if the code is run twice on the same computer using the same value of t, the same value of $y_0(t)$ will result.

We are specifically interested here in computer models that can provide not only the response $y_0(t)$ but also first partial derivatives $y_j(t) = \partial y_0(t)/\partial t_j$, $j = 1, 2, \ldots, k$. Development of this capability has been inspired by a strong scientific interest in identifying the inputs that have the greatest (or least) effect on the response. Under one approach, a system of *adjoint equations* for the partial derivatives of a response with respect to a set of inputs is formulated and solved along with the original model equations. This has been implemented, for example, in the LEAP-78 energy-economics model described by Alsmiller et al. (1983), a design model of a large liquid-metal fast breeder reactor described by Marable, Weisbin, and de Saussure (1980), and a radiative-convective climate model described by Hall, Cacuci, and Schlesinger (1982). A second approach uses automatic differentiation. Research in this area has produced computer-automated methods for "enhancing" computer codes—that is, expanding existing codes that compute only outputs so that they also compute derivatives (e.g., Griewank 1989; Oblow, Pin, and Wright 1986; Worley, Wright, Pin, and Harper 1986).

Here we are interested in using derivative information for the prediction of $y_0(t)$ at points $t \in T$ that have not been directly observed. This is motivated by applications requiring many evaluations of y_0, such as numerical optimization or uncertainty analysis, in

which repeated execution of the model may be prohibitive due to computing expense. Hence we seek to develop a *fast predictive approximation* to y_0, one that is sufficiently accurate for the desired purpose, based on relatively few actual runs.

We shall do this within the framework of Bayesian prediction, using a class of random functions (stochastic processes, random fields) to express uncertainty about the function y_0. Currin, Mitchell, Morris, and Ylvisaker (1991) described various implementations of this, for computer experiments in which the model evaluates $y_0(t)$ but not its derivatives. A parallel approach, related to the spatial modeling techniques of kriging, was described by Sacks, Schiller, and Welch (1989) and Sacks, Welch, Mitchell, and Wynn (1989). In both of these approaches, the values of y_0 generated by the computational model were regarded as "data" that, unlike most physical measurements, are exactly reproducible. In this article, these data also include derivatives.

Our approach to Bayesian prediction using derivatives is outlined in Section 1, and the mechanics are demonstrated by means of a simple example in Section 2. We discuss the experimental design problem—how the values of t can be chosen for the needed runs of the computer model—in Section 3. Our emphasis here is on D-optimal designs based on weak prior information. The example problem is extended and continued with an examination of fast predictive approximations based on data from several different types of designs in Section 4. A discussion, having to do mainly with cost-benefit issues, is given in Section 5.

1. METHODOLOGY

We represent prior uncertainty about the unknown function $y_0(t)$, $t \in T$, by the Gaussian stochastic process $Y_0 = \{Y_0(t), t \in T\}$, with mean function

$$\mu_0(t) = E[Y_0(t)] \tag{1.1}$$

and positive definite covariance function

$$K_{00}(t, s) = \text{cov}[Y_0(t), Y_0(s)]. \tag{1.2}$$

This means that, for every finite set $S = \{s^1, s^2, \ldots, s^m\}$ in T, prior uncertainty about the response vector $(y_0(s^1), y_0(s^2), \ldots, y_0(s^m))^T$ is represented by the multivariate normal random vector $(Y_0(s^1), Y_0(s^2), \ldots, Y_0(s^m))^T$, with mean and covariance matrix determined from (1.1) and (1.2).

The specification of a prior process, with appropriate mean and covariance functions, determines also various derivative processes. (See Parzen [1962, p. 83] for formal definitions and conditions for existence.) Let

$$Y_j(t) = \frac{\partial Y_0(t)}{\partial t_j}$$

$$= \lim_{h \to 0} \frac{Y_0(t_1, \ldots, t_j + h, \ldots, t_k) - Y_0(t_1, \ldots, t_j, \ldots, t_k)}{h} \tag{1.3}$$

and

$$y_j(t) = \frac{\partial y_0(t)}{\partial t_j}, \tag{1.4}$$

where $j = 1, 2, \ldots, k$. The uncertainty about $y_j(t)$ is expressed by the derivative process $Y_j(t)$.

The general derivative process

$$Y_0^{(a_1, a_2, \ldots, a_k)} = \frac{\partial^{a_1 + a_2 + \cdots + a_k}}{\partial t_1^{a_1} \partial t_2^{a_2} \cdots \partial t_k^{a_k}} Y_0,$$

$$a_j \geq 0, j = 1, 2, \ldots, k, \tag{1.5}$$

is Gaussian (since Y_0 is Gaussian), with mean function and covariance function given by

$$E[Y^{(a_1, a_2, \ldots, a_k)}(t)] = \mu_0^{(a_1, a_2, \ldots, a_k)}(t) \tag{1.6}$$

and

$$\text{cov}[Y_0^{(a_1, a_2, \ldots, a_k)}(t), Y_0^{(b_1, b_2, \ldots, b_k)}(s)]$$

$$= K_{00}^{(a_1, \ldots, a_k, b_1, \ldots, b_k)}(t, s). \tag{1.7}$$

In this article, we consider the situation in which the functions $y_0, y_1, \ldots, y_k$ are observed at the set of design *sites* (points) $D = \{t^1, t^2, \ldots, t^n\}$. We organize these data in the $n(k + 1)$-vector $\vec{y}$ as n successive segments of length $k + 1$:

$$\vec{y} = (y_0(t^1), \ldots, y_k(t^1), y_0(t^2), \ldots, y_k(t^n))^T. \tag{1.8}$$

Prior uncertainty about $\vec{y}$ is represented by the random normal $n(k + 1)$-vector

$$\vec{Y} = (Y_0(t^1), \ldots, Y_k(t^1), Y_0(t^2), \ldots, Y_k(t^n))^T \tag{1.9}$$

with mean vector $\vec{\mu}$ and covariance matrix Σ obtained via (1.6) and (1.7), respectively.

We shall consider here only covariance functions $K_{00}(t,s)$ for which Σ is positive definite for any design D composed of distinct sites. This is true of the covariance functions given by Sacks, Welch, Mitchell, and Wynn (1989) and Currin et al. (1991), as it is for many of the covariance functions used in spatial statistics.

Application of standard formulas for conditional multivariate normal distributions shows that the (Gaussian) posterior process $Y_0^* = \{Y_0^*(t), t \in T\}$ has mean function

$$\mu_0^*(t) = E[Y_0^*(t)]$$

$$= \mu_0(t) + (\vec{y} - \vec{\mu})^T \Sigma^{-1} \vec{\kappa}(t), \tag{1.10}$$

where $\vec{\kappa}(t)$, the vector of covariances of $Y_0(t)$ with $\vec{Y}$, is obtained using (1.7).

The posterior covariance function is

$$K_{00}^*(t, s) = \text{cov}[Y^*(t), Y^*(s)] = K_{00}(t, s) - \vec{\kappa}^T(t)\,\Sigma^{-1}\vec{\kappa}(s). \quad (1.11)$$

Following execution of the computer model at each site in design D, we use the posterior mean $\mu_0^*(t)$ as a fast predictive approximation for the true response $y(t)$ at any site t and the posterior standard deviation $\sigma_0^*(t) = \sqrt{K_{00}^*(t, t)}$ as a measure of the uncertainty of prediction there.

The specification of the prior is the central issue in practice. As in the work of Currin et al. (1991), we simplify matters by adopting various *stationarity* restrictions:

$$\mu_0(t) = \mu \quad (1.12)$$

and

$$K_{00}(t, s) = \sigma^2 R(s_1 - t_1, \ldots, s_k - t_k), \quad (1.13)$$

where R is a correlation function that depends only on the differences between s and t in each coordinate. Further simplification comes from adoption of the *product correlation rule*:

$$R(s_1 - t_1, \ldots, s_k - t_k) = \prod_{j=1}^{k} R_j(s_j - t_j), \quad (1.14)$$

where R_j's are chosen from a parametric family of suitably differentiable correlation functions on the real line.

Then $\mu_j(t) = E[Y_j(t)] = 0$ if $j \geq 1$ and (1.7) may be written as

$$\text{cov}[Y_0^{(a_1,a_2,\ldots,a_k)}(t), Y_0^{(b_1,b_2,\ldots,b_k)}(s)] = \sigma^2(-1)^{\Sigma a_j} \prod_{j=1}^{k} R_j^{(a_j+b_j)}(s_j - t_j). \quad (1.15)$$

Note that, since we are dealing with (at most) first partial derivatives, each a_j and b_j is 0 or 1. Moreover, simplified versions of (1.15) occur for $t = s$, since $R_j(0) = 1$ and $R_j'(0) = 0$, $j = 1, \ldots, k$.

Of course, the chosen R_j's must correspond to processes that are at least once differentiable; this means that each R_j must be twice differentiable. [This is clear from (1.15); also see Parzen (1962, p. 84).] Gaussian processes with the correlation function used by Sacks, Welch, Mitchell, and Wynn (1989),

$$R_j(s_j - t_j) = e^{-\theta_j|s_j - t_j|^{\alpha_j}}, \quad (1.16)$$

with $\theta_j > 0$ and $0 < \alpha_j \leq 2$, are infinitely differentiable for $\alpha_j = 2$ but not differentiable at all for $\alpha_j < 2$. In his discussion of that article, Stein (1989) referred to an alternative class of processes that is exactly m times differentiable, $m > 1$. A useful way to derive differentiable processes is by integrating known processes—see Mitchell, Morris, and Ylvisaker (1990) for some examples that are stationary on an interval. In the examples of this article, we shall use (1.16) with $\alpha_j = 2$.

Under our stationarity restrictions, the prior mean of $\vec{Y}$ is

$$\vec{\mu} = \mu\vec{v}, \quad (1.17)$$

where $\vec{v}$ is a binary vector with 1 in position $(i - 1)(k + 1) + 1$, $i = 1, \ldots, n$—that is, in each position corresponding to the mean of some $Y_0(t^i)$—and 0 everywhere else. Moreover, the prior covariance matrix of $\vec{Y}$ is

$$\Sigma = \sigma^2 C \quad (1.18)$$

and the vector of prior covariances between $Y_0(t)$ and $\vec{Y}$ is

$$\vec{\kappa}(t) = \sigma^2\vec{r}(t), \quad (1.19)$$

where C and $\vec{r}(t)$, which do not depend on σ^2, can be obtained using (1.15) with $\sigma^2 = 1$.

Then the mean of the posterior process $Y^*(t)$ at (1.10) becomes

$$\mu_0^*(t) = \mu + (\vec{y} - \mu\vec{v})^T C^{-1}\vec{r}(t), \quad (1.20)$$

and the posterior covariance function at (1.11) becomes

$$K_{00}^*(t, s) = \sigma^2(1 - \vec{r}^T(t)C^{-1}\vec{r}(s)). \quad (1.21)$$

These expressions require specification of the scalars μ and σ and the correlation functions R_j. In practice, we choose a parametric family for each R_j a priori but allow its parameters [e.g., θ_j in (1.16)], and also μ and σ, to be determined by the data, usually by maximum likelihood.

The log-likelihood is, apart from additive and multiplicative constants,

$$L(\mu, \sigma, \theta) = -n(k + 1)\ln\sigma^2 - \ln|C_\theta| - \frac{1}{\sigma^2}(\vec{y} - \mu\vec{v})^T C_\theta^{-1}(\vec{y} - \mu\vec{v}), \quad (1.22)$$

where dependence on the correlation parameters, collectively denoted as θ here, is now explicitly indicated. For fixed θ, maximization of L over μ and σ^2 is obtained by

$$\hat{\mu}_\theta = \frac{\vec{v}^T C_\theta^{-1}\vec{y}}{\vec{v}^T C_\theta^{-1}\vec{v}} \quad (1.23)$$

and

$$\hat{\sigma}_\theta^2 = \frac{1}{n(k+1)}(\vec{y} - \hat{\mu}_\theta\vec{v})^T C_\theta^{-1}(\vec{y} - \hat{\mu}_\theta\vec{v}). \quad (1.24)$$

Determination of $\hat{\theta}$, which requires maximization of $L(\hat{\mu}_\theta, \hat{\sigma}_\theta, \theta)$, is usually done by constrained iterative

search. Although this can be done using routines from standard mathematical software libraries, it may require a considerable amount of computation, depending on the dimension of θ and the values of n and k.

Generalization to the case in which μ and σ have the usual *noninformative* prior distributions—that is, μ and $\log \sigma$ have independent improper uniform prior distributions—is relatively straightforward, but a fully Bayesian approach, in which vague priors are also attached to the correlation parameters, appears difficult to implement.

2. EXAMPLE: BOREHOLE MODEL

In his discussion of a method of uncertainty analysis, Worley (1987) used a simple demonstration model of the flow of water through a borehole that is drilled from the ground surface through two aquifers. [His use of this particular model follows that of Harper and Gupta (1983), who used it in demonstrating other methods of uncertainty analysis.] The model formulation is based on assumptions of no groundwater gradient, steady-state flow from the upper aquifer into the borehole and from the borehole into the lower aquifer, and laminar, isothermal flow through the borehole. The response variable Worley examined from this model is y_0, the flow rate through the borehole in m^3/yr, which is determined by the equation

$$y_0 = \frac{2\pi T_u(H_u - H_l)}{\ln(r/r_w)\left[1 + \dfrac{2LT_u}{\ln(r/r_w)r_w^2K_w} + \dfrac{T_u}{T_l}\right]}, \quad (2.1)$$

where the eight inputs and their respective ranges of interest and units are as follows:

r_w = radius of borehole, .05 to .15 m
r = radius of influence, 100 to 50,000 m
T_u = transmissivity of upper aquifer, 63,070 to 115,600 m^2/yr
H_u = potentiometric head of upper aquifer, 990 to 1,110 m
T_l = transmissivity of lower aquifer, 63.1 to 116 m^2/yr
H_l = potentiometric head of lower aquifer, 700 to 820 m
L = length of borehole, 1,120 to 1,680 m
K_w = hydraulic conductivity of borehole, 9,855 to 12,045 m/yr

Since y_0 can be expressed as a simple, explicit equation in the inputs, this function certainly is not typical of the computationally intensive computer models that motivate this work. It is useful for demonstration purposes, however, since its simplicity will allow us to quickly assess the accuracy of predictions at many test sites via direct evaluation. In Section 4, we shall demonstrate the use of our methodology for predicting y_0 as a function of all eight inputs. Here, to illustrate the mechanics of the required calculations, we shall consider only two, r_w and K_w, and fix the remaining inputs at their respective lowest values. The range of K_w has been extended (for this exercise only) to [1,500, 15,000] to produce a somewhat more nonlinear, nonadditive function. Moreover, the two input variables considered here have been scaled so that each takes its values from the unit interval; the scaled versions of r_w and K_w are denoted by t_1 and t_8, respectively. Figure 1 is a contour graph of y_0 as a function of t_1 and t_8 over the region of interest.

For demonstration purposes, consider the experimental design at the three sites marked as heavy dots on Figure 1. The data, y_0 and its first derivatives with respect to t_1 and t_8 are displayed in Table 1. We place these values in the data vector $\vec{y}$, as indicated at (1.8):

$$\vec{y} = (3.0489, 12.1970, 27.4428, 71.6374, \ldots, 244.4854)^T. \quad (2.2)$$

The prior covariance matrix Σ at (1.18) is organized as an $n \times n = 3 \times 3$ array of $(k + 1) \times (k + 1) = 3 \times 3$ blocks, where the ith diagonal block holds the within-site covariances at t^i and the (i, j)th off-diagonal block holds the between site covariances

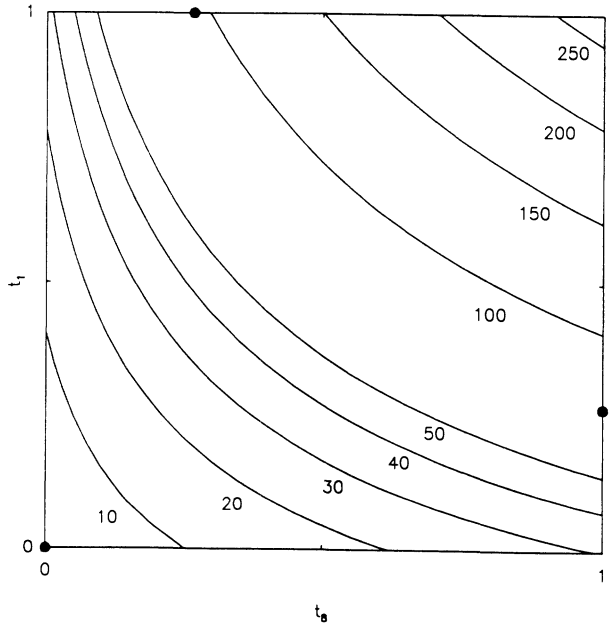

Figure 1. Contours of Output y_0 as a Function of t_1 and t_8 in the Borehole Example of Section 2.

Table 1. Design and Data for a Simple Example

Site	t_1	t_8	y_0	$y_1 = \partial y_o/\partial t_1$	$y_8 = \partial y_o/\partial t_8$
t^1	.0000	.0000	3.0489	12.1970	27.4428
t^2	.2680	1.0000	71.6374	185.7917	64.1853
t^3	1.0000	.2680	93.1663	123.6169	244.4854

corresponding to the pair (t^i, t^j); that is,

$$\Sigma = \sigma^2 C = \sigma^2 \begin{bmatrix} C^{11} & C^{12} & C^{13} \\ C^{21} & C^{22} & C^{23} \\ C^{31} & C^{32} & C^{33} \end{bmatrix},$$

where, for example,

$$C^{12} = \sigma^{-2} \begin{bmatrix} K_{00}(t^1, t^2) & K_{01}(t^1, t^2) & K_{08}(t^1, t^2) \\ K_{10}(t^1, t^2) & K_{11}(t^1, t^2) & K_{18}(t^1, t^2) \\ K_{80}(t^1, t^2) & K_{81}(t^1, t^2) & K_{88}(t^1, t^2) \end{bmatrix},$$

and where $K_{pq}(t^1, t^2) = \text{cov}[Y_p(t^1), Y_q(t^2)]$ can be computed using (1.15) with $\sigma^2 = 1$. [Recall that $Y_0(t) = Y_0^{(00000000)}(t)$, $Y_1(t) = Y_0^{(10000000)}(t)$, and $Y_8(t) = Y_0^{(00000001)}(t)$.] For example, the (1, 3) and (2, 2) elements of C^{12} are, respectively, $\sigma^{-2}K_{08}(t^1, t^2) = R_1(t_1^2 - t_1^1)R_8'(t_8^2 - t_8^1) = R_1(.2680)R_8'(1)$ and $\sigma^{-2}K_{11}(t^1, t^2) = -R_1''(t_1^2 - t_1^1)R_8(t_8^2 - t_8^1) = -R_1''(.2680) \times R_8(1)$, where we use the correlation function (1.16) with $\alpha_j = 2$, for which $R_j(x) = e^{-\theta j x^2}$, $R_j'(x) = -2\theta_j x e^{-\theta j x^2}$, and $R_j''(x) = (-2\theta_j + 4\theta_j^2 x^2)e^{-\theta j x^2}$.

Now we maximize the log-likelihood (1.22), using a standard numerical optimization routine in conjunction with (1.23) and (1.24), noting also that $\vec{v} = (1\,0\,0\,1\,0\,0\,1\,0\,0)^T$ here. The greatest log-likelihood occurs at $\hat{\theta}_1 = .429$ and $\hat{\theta}_8 = .467$; the corresponding maximum likelihood values for μ and σ are $\hat{\mu} = 69.15$ and $\hat{\sigma} = 135.47$.

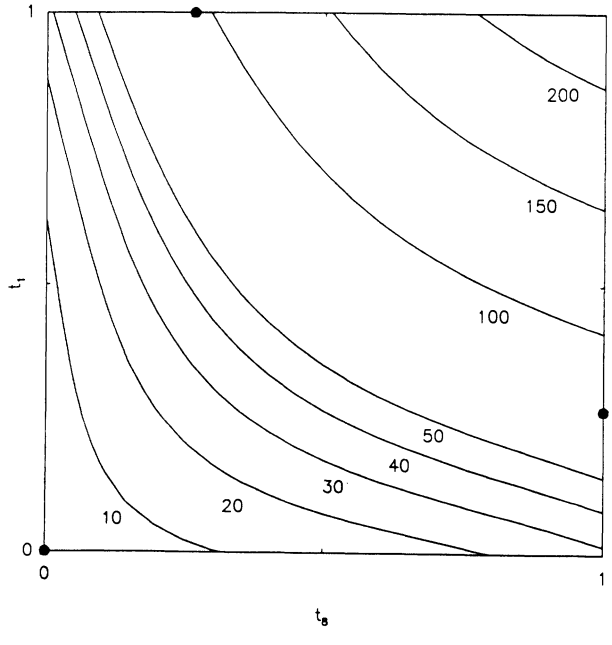

Figure 2. Contours of Predicted Output $\hat{y}_o$ as a Function of t_1 and t_8 in the Borehole Example of Section 2.

To calculate the posterior mean and variance of $Y_0(t)$ at an arbitrary site t, we set θ, μ, and σ to their maximum likelihood values in (1.20) and in (1.21) with $t = s$. Note that the nine-vector $C^{-1}(\vec{y} - \mu\vec{v})$ is already available from the computation of (1.24) needed by the maximum likelihood algorithm. The nine-vector $\vec{r}(t) = \sigma^{-2}\vec{\kappa}(t) = \sigma^{-2}(K_{00}(t, t^1), K_{01}(t, t^1), K_{08}(t, t^1), \ldots, K_{00}(t, t^3), K_{01}(t, t^3), K_{08}(t, t^3))^T$ can be computed using (1.15) with $\sigma^2 = 1$ and $\theta = \hat{\theta}$. This is similar to the computation of the first column of C and in fact is identical to the first column of C if $t = t^1$.

Once $\vec{r}(t)$ is determined, the posterior mean (1.20) amounts to little more than the inner product of two nine-vectors. The computation of the posterior variance is much more expensive, however, since the solution of the 9×9 system $Cx = \vec{r}(t)$ is required for every prediction site t. One should, of course, take advantage of the fact that the matrix C of the linear system is the same for all t; that is, what is required here is an efficient solver for linear systems with multiple right sides.

Here we find, for example, that the posterior mean at $t = (.5, .5)$ is 69.4 with a posterior standard deviation of 2.7. At $t = (1, 1)$, the posterior mean is 230.0 and the posterior standard deviation is 19.2. Predictions on a 21×21 grid were generated in this way and used to produce the contour graph of $\hat{y}_0(t_1, t_8)$ over the region of interest, as shown in Figure 2.

3. OPTIMAL DESIGN

An advantage to the use of stochastic processes as priors for y_0 is that the variability of the posterior process Y_0^*, as expressed by the posterior covariance function K_{00}^* at (1.21), can be used to provide measures of uncertainty, and designs can be sought to minimize the expected uncertainty in some sense. See Ylvisaker (1987) and Sacks, Welch, Mitchell, and Wynn (1989) for references to some previous work along these lines. Criteria that have been considered are G optimality (minimization of the maximum variance of $\{Y_0^*(s), s \in T\}$), A optimality (minimization of the average variance of $\{Y_0^*(s), s \in T\}$), and D optimality (minimization of the determinant

of the covariance matrix of $\{Y_0^*(s), s \in S\}$, where $S = \{s^1, s^2, \ldots, s^m\}$ is a chosen finite subset of T).

Johnson, Moore, and Ylvisaker (1990) established an interesting link between these criteria and the geometric properties of certain designs for the case in which only the response is observed at each design site. One of their results implies that, when T is finite and the prior correlation between sites is extremely weak and is a positive decreasing function of an appropriately defined intersite distance $d(t^i, t^j)$, a design $D = \{t^1, t^2, \ldots, t^n\}$ is D-optimal (for $S = T - D$) among all feasible n-run designs only if (a) the minimum intersite distance $\underline{d}(D) = \min_{i,j} d(t^i, t^j)$ is maximized and (b) the *index* $J(D)$—that is, the number of pairs (i, j), $i < j$ for which $d(t^i, t^j) = \underline{d}(D)$—is minimized. Designs that satisfy these conditions will be called *maximin distance designs of minimum index* (or simply *maximin designs*).

Similar results can be obtained in the case in which both the response and its first partial derivatives are observed. Let $\overline{T}$ be a finite set of sites in T, from which the design set D is to be selected—we will observe $y_0, y_1, \ldots, y_k$ on D. To minimize the determinant of the covariance matrix of $\{Y_0^*(s), Y_1^*(s), \ldots, Y_k^*(s), s \in \overline{T} - D\}$, it suffices to maximize the determinant of Σ, the prior covariance matrix of $\vec{Y}$. This follows from the main result of Shewry and Wynn (1987). In the present setting, where our stated goal is to predict y_0 and not necessarily its derivatives, a more appropriate criterion would be to minimize the determinant of the covariance matrix of $\{Y_0^*(s), s \in \overline{T} - D\}$. For reasons that are too detailed to warrant describing here, this does not lend itself to a manageable design procedure, so we shall adopt the maximization of $|\Sigma|$ as our criterion, under the assumption that designs that are optimal for predicting y_0 and its derivatives should be good for predicting y_0 alone. We would normally take $\overline{T}$ to be a very large, dense grid, so knowledge of y_0 everywhere on $\overline{T}$ should be very nearly equivalent to knowledge of y_0 and its first derivatives everywhere in T.

Having settled on maximization of $|\Sigma|$, a result similar to that of Johnson et al. (1990) can be obtained. Specifically, let $K_{00}(t, s) = e^{-\theta d^2(t,s)}$, where $d^2(t, s) = \Sigma(s_j - t_j)^2$ is the squared Euclidean distance between sites t and s; K_{00} is just the product correlation that corresponds to (1.16) with $\theta_j = \theta$ and $\alpha_j = 2$. Then, a necessary condition for a design to be D optimal as $\theta \to \infty$ is that it be a maximin (Euclidean) distance design of minimum index. The proof of a somewhat more general version of this result, applicable to any correlation of the product form (1.14), is given in the Appendix.

When constructing a design in practice, of course, the particular correlation function that will ultimately be chosen for the analysis is unknown. We intuitively tend to favor designs based on very weak prior information. This leads to consideration of maximin designs, although the limiting (weak) correlation needed to link the maximin property to D optimality is not useful for analysis.

To construct specific maximin designs, we wrote a computer program to find designs that minimize a surrogate criterion function:

$$\phi_p(D) = \left[\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} d_{ij}^{-p}(D)\right]^{1/p}, \qquad (3.1)$$

where $d_{ij}(D) = d(t^i, t^j)$ is the Euclidean distance between the ith and jth design sites in D. To see the motivation for this, first rewrite (3.1) as

$$\phi_p(D) = \frac{1}{\underline{d}(D)}\left[\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(\frac{\underline{d}(D)}{d_{ij}(D)}\right)^p\right]^{1/p},$$

where $\underline{d}(D)$ is the smallest Euclidean distance between any two sites in D. For pairs of sites separated by this distance, the corresponding term in the sum is 1. If a large value is chosen for p, pairs of sites separated by greater distances will have associated terms in the sum that are approximately 0. Hence, for large p,

$$\phi_p(D) \approx \frac{J^{1/p}(D)}{\underline{d}(D)},$$

where $J(D)$ is the index of D. For large enough p, minimizing ϕ_p is primarily accomplished by maximizing $\underline{d}$ and to a much smaller degree by minimizing J.

Our computer program for minimizing ϕ_p implements a simple point-exchange algorithm based on the optimization technique of simulated annealing. [See Kirkpatrick, Gelatt, and Vechhi (1983) for a discussion of simulated annealing, or Bohachevsky, Johnson, and Stein (1986) for a generalization of this technique applied to a statistical problem.] In our application, a search begins with a randomly constructed design, which is sequentially modified as follows. First, one site from the current design is randomly selected, and each coordinate of that site is subjected to a trial random perturbation. (Specific distributions of perturbations used, and other particulars of the search, are given for the example application of Sec. 4.) Suppose that the perturbation changes the value of ϕ_p by an amount $\Delta\phi_p$. If $\Delta\phi_p$ is negative—that is, the perturbation improves the design criterion—the perturbed design is accepted, and it becomes the current design. On the other hand, if $\Delta\phi_p$ is nonnegative, the perturbed design is accepted only with probability $P_a(\Delta\phi_p)$, which decreases as $\Delta\phi_p$ increases. During the course of the search, the algorithm keeps track of the best design found; this will not necessarily be the same as the

current design at any iteration, since the design criterion may increase from one iteration to another. If the best design is unchanged for a specified number of iterations at a given $P_a(\Delta\phi_p)$, a decision is made to lower $P_a(\Delta\phi_p)$ (by a predetermined factor) or to stop the search. The latter decision is made only if no perturbations at the current level of $P_a(\Delta\phi_p)$ produced a negative $\Delta\phi_p$. The best design found during the course of the search is then reported.

4. EXAMPLE REVISITED

We now return to our example model, described by Equation (2.1), to demonstrate an application of this method for an eight-dimensional input vector. For this purpose, all eight inputs were scaled as in Section 2 so that the range of each t_j was the unit interval and $T = [0,1]^8$. Initially, we applied the prediction method to the design used by Worley (1987) in his demonstration of a methodology he calls "deterministic uncertainty analysis," which uses both observed values of y_0 and its first derivatives. Although his primary interest was in exploring how a specified probability distribution on t is propagated to y_0, his analysis included an interim step that involves prediction of y_0 at unobserved sites, using local first-order Taylor series expansions of y_0. In his demonstration, Worley's experimental design was a 10-run Latin hypercube sample (McKay, Conover, and Beckman 1979), generated using a nonuniform distribution across T, and he compared predictions of y_0 with its actual value at sites in a 50-run Latin hypercube test set, generated using the same distribution. He reported root mean squared errors over these 50 sites of 1.89, 2.45, and 2.37 for three versions of his method; for comparison, the range of true values of y_0 over the test set is 24.97 to 144.57. Our prediction procedure, when applied to the same experimental data, produced predictions having a root mean squared error of .610 over the same set of 50 test sites. Encouraged by this result, we undertook the more extensive investigation that we describe in this section.

We tried four different experimental designs, each having 10 runs. We refer to our first design as a *Latin hypercube design* because of its basic structure, although it was generated in a manner that differs from the original method proposed by McKay et al. (1979). For each input variable, we selected 10 equally spaced levels in [0,1], including the extremes 0 and 1. Each of the eight columns of the 10 × 8 design matrix was generated by randomly permuting these 10 levels. [This is quite similar to the *lattice sampling designs* discussed by Patterson (1954).] Although we are not fitting linear models here, we thought there might be some advantage to avoiding highly correlated columns in the design matrix, so we generated 100 independent Latin hypercubes in the manner indicated and selected the one that minimized the largest R^2 resulting from the regression of any one column of the design matrix on the others. [An alternative method of generating Latin hypercube samples with small correlations was described by Owen (1990).]

Our second design was a *maximin design* in 10 runs, generated using the algorithm described in Section 3. After some initial experimenting with the algorithm to find annealing parameters that appeared to be effective for this problem, our first attempt at finding an optimal design consisted of 10 searches, in which each element of the starting design matrix was chosen randomly from the unit interval, perturbations were normally distributed with a standard deviation of .3 (except when this would result in a value outside the unit interval, in which case the change was modified to yield either 0 or 1), and p = 1,000. Five of these searches resulted in designs with $\underline{d} = 2$, and the other five produced smaller values. Of the five with $\underline{d} = 2$, one had an index of 42, three had indexes of 38, and one had an index of 37. These five designs (unlike the others) also placed all sites in the corners of T.

Following this last observation, 10 additional searches were made using a modified search in which only designs on the 2^8 corners of T were considered; that is, each coordinate in the initial design was 0 or 1 with equal probability. Here, once a site was selected for modification, the level of an individual coordinate was reversed with probability .3; again p = 1,000. Of these searches, nine yielded designs with $\underline{d} = 2$, one of these with index 40 and the remaining eight with index 36. Our (tentative) conclusion based on this search was that these last eight designs are optimal, although they are not all equivalent. The design we chose from this set is given in Table 2.

A potential strength of the Latin hypercube design is that it has very good one-dimensional projections; that is, projection of the design onto any coordinate yields an equispaced 10-level design (which is, in fact,

Table 2. A Maximin Distance Design in $[0,1]^8$ for $n = 10$ ($\underline{d} = 2$, $J = 36$)

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
1	1	0	0	1	0	1	1
1	1	1	1	0	0	1	0
1	0	0	1	1	0	0	0
0	1	0	0	1	1	0	0
1	1	0	1	0	1	0	1
0	1	1	0	0	0	0	1
0	0	1	1	1	0	1	1
0	0	0	0	0	1	1	1
0	0	1	1	0	1	0	0
1	0	1	0	1	1	1	0

the maximin design in one dimension). Two-dimensional projections also cover the $[0,1]^2$ square fairly well, partly because of our effort to choose a Latin hypercube with low correlations among design columns. By contrast, the maximin design has very bad one-dimensional projections here, since each input variable takes on only two levels.

On the other hand, the generation of the Latin hypercube design essentially ignores the geometrical relationships among the design sites in the full eight-dimensional space, whereas the maximin design considers these relationships explicitly. Our third and fourth designs are *compromise designs* that seek to capture the advantages of both.

The first compromise, which we shall call a *maximin Latin hypercube design*, results from applying the maximin criterion within the class of Latin hypercube designs. [Park (1991) also considered design optimization within the class of Latin hypercube designs, although he did not consider maximin designs.] The construction was based on our simulated annealing algorithm with minor modifications. A randomly constructed Latin hypercube was used as the starting design in each search, and trial perturbations were created by exchanging two entries in a randomly chosen column of the design matrix; the result of any such exchange is another Latin hypercube. Although 20 searches were attempted, the apparent maximin design in this class was generated only once, so it is quite possible that the result is not a true optimum.

The second compromise, which we shall call the *modified maximin design*, is a modification of our maximin design to give it the desired one-dimensional projections. Starting with each column of the design matrix of Table 2, the five 0s were replaced with the values 0, 1/9, 2/9, 3/9, and 4/9, assigned in random order, and the five 1s were similarly replaced with 5/9, 6/9, 7/9, 8/9, and 1. This procedure is essentially the same as randomly selecting a Latin hypercube design from among those which, if each entry were rounded to 0 or 1, would be the maximin design given in Table 2. (Tang [in press] similarly considered random choice of Latin hypercube designs with the restriction that rounding each entry produces an orthogonal array.)

For each of the four designs just described, the value of y_0 and its first derivatives were generated at each design site. Predictions were made as described in Sections 1 and 2, where the correlation function R is given by (1.14) and (1.16) with $\alpha_j = 2$. In each case, the parameters μ, σ, and θ_j ($j = 1, 2, \ldots, 8$) were estimated by the method of maximum likelihood, using a version of the Nelder–Mead simplex algorithm. The search was restricted to the region $.01 \le \rho_j \le .99$ ($j = 1, \ldots, 8$), where $\rho_j = R_j(1) = e^{-\theta_j}$. The maximum likelihood values for the ρ_j's are given in Table 3.

Convergence to the maximum likelihood values was rather slow (about 700 iterations in a typical case) and required occasional intervention to fix some of the ρ_j's at their boundaries or to release those that had previously been fixed. In one case, we also tried an arctangent transformation to transform the range of each ρ_j from (.01, .99) to $(-\infty, \infty)$. The hope was to avoid difficulties at the boundary, but the algorithm tended to bog down short of the solution, and intervention was again used to periodically redefine a set of ρ_j's to be fixed at a boundary. This method too took about 700 iterations. Fortunately, we had access to a fairly fast computer, which was able to do a single iteration (requiring the solution of a 90×90 linear system) in 1.5 seconds. Although we do not mean to understate the amount of computation that is needed, it is clear that a more sophisticated constrained optimization routine would decrease the number of function evaluations considerably.

To evaluate how well our predictions matched the true model, we selected two "test sets" of sites at which to compare $\hat{y}_0$ and y_0. The first of these is a random sample of 400 sites, selected from the uniform distribution over T. The second set of test sites is the 256 corners of T—that is, those sites at which each of the inputs takes either the high or low extreme value in its range. The first set is intended to provide an indication of how well each predictor does throughout the interior of T, while the second allows us to compare their performance at the extreme sites. Values of y_0 range from 12.4035 to 230.6478 in the first test set and from 7.8197 to 309.5756 in the second. Predictions were made for each design, at the sites in each test set, and errors of prediction ($\hat{y}_0 - y_0$) were calculated. These errors are summarized by the boxplots in Figures 3 and 4. The Latin hypercube

Table 3. Maximum Likelihood Estimates for the Example of Section 4.

	Design			
Parameter	Latin hypercube	Maximin	Maximin LH	Modified maximin
μ	124.28	106.99	110.44	128.72
σ	90.58	131.17	111.46	120.65
ρ_1	.35	.49	.48	.53
ρ_2	.69	.73	.83	.99
ρ_3	.99	.99	.99	.99
ρ_4	.90	.97	.98	.97
ρ_5	.99	.99	.99	.99
ρ_6	.96	.96	.98	.95
ρ_7	.94	.81	.89	.83
ρ_8	.99	.99	.99	.98

NOTE: $R(s - t) = \Pi_j \rho_j^{(s_j - t_j)^2}$.

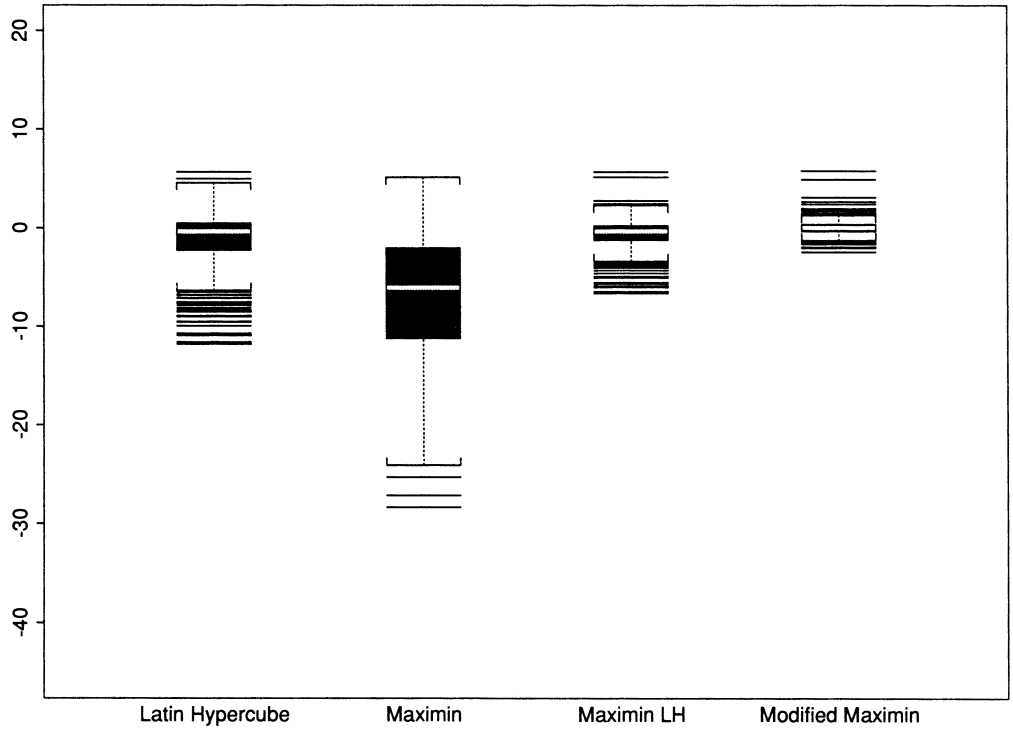

Figure 3. Prediction Errors at 400 Random Test Sites, for Four Different Designs.

appears to be superior to the maximin design on the random test set, but the reverse is true on the corners of T. Overall, the two compromise designs seem to do better, particularly the modified maximin design.

We were not able to discern a clear connection between prediction performance and the distance between the test sets and the design sets. In the 400-site test set, the maximum distance to the design set was 1.11 for the maximin Latin hypercube, 1.14 for the Latin hypercube, 1.17 for the modified maximin

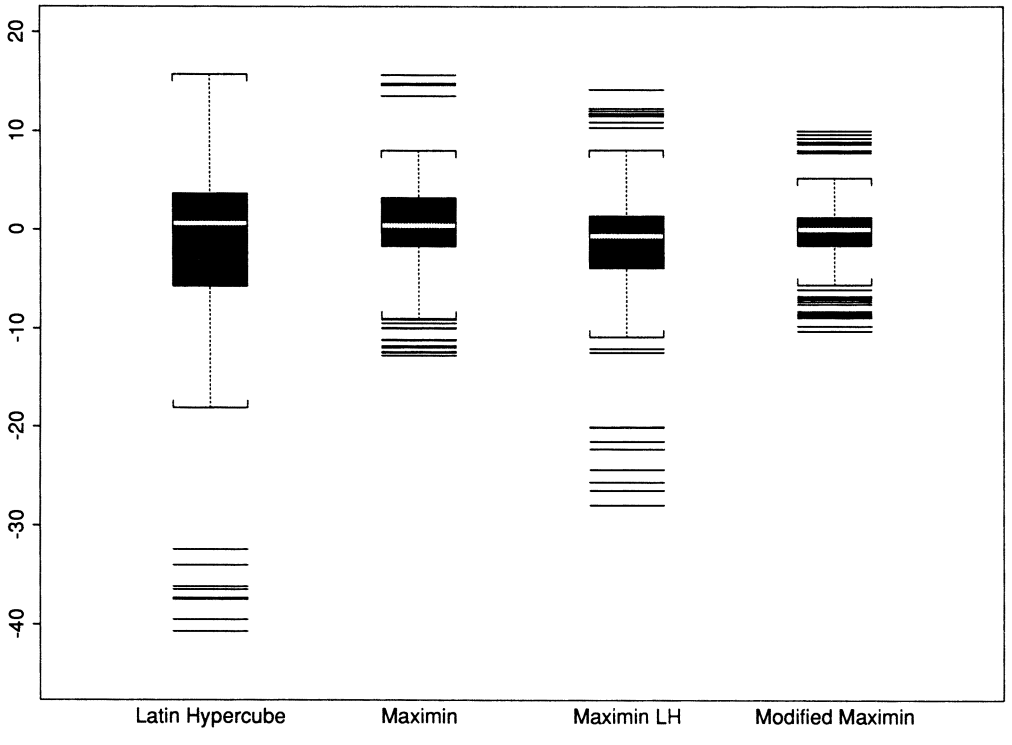

Figure 4. Prediction Errors at the 256 Corner Sites, for Four Different Designs.

design, and 1.45 for the maximin design. In the 256-run test set, the maximum distance to the design set was 1.45 for the maximin Latin hypercube, 1.48 for the Latin hypercube, 1.55 for the modified maximin design, and 1.73 for the maximin design.

Although we shall not go beyond the construction of the prediction surface $\hat{y}_0$ here, it is appropriate to mention that $\hat{y}_0$ can be examined in various ways to gain insight into the nature of the importance of the inputs and their interactions. A useful approach, for example, is the display of main effect and interaction functions suggested by Sacks, Welch, Mitchell, and Wynn (1989, p. 418) and used by Welch et al. (1992). Prediction using derivatives thereby provides an effective method of synthesizing the fragmentary information obtained in a sensitivity analysis.

5. DISCUSSION

In practice, it will be important to consider issues of computational cost versus benefit when deciding whether and how to use derivatives in a computer experiment. Although these issues are heavily application dependent, a few general observations may be made.

For the purpose of prediction using the methods of this article, we think there is nothing to be gained (and probably much to be lost) by designing an experiment to produce derivatives by the method of divided differences. This claim is supported by a 90-run variation of the borehole experiment (without derivatives) that we shall describe later in this section. In this article, however, we have assumed that derivatives at sites in T are already available (perhaps having been generated as part of a sensitivity analysis), or that the capability for generating them is available. In the first situation, one would expect to achieve better predictions by using the derivative information than by ignoring it. To see how much was gained by using the derivatives in the example of Section 4, we repeated the example for a couple of the designs, using only the observed y_0 at each of the 10 design sites, with the same type of correlation function. For the Latin hypercube design, the maximum absolute errors and root mean squared errors were roughly four times larger than those found when the derivatives were used. For the modified maximin design, the increase in magnitude of errors was roughly tenfold.

The cost of using available derivatives is primarily the computational cost of estimating θ—that is, repeated evaluation of the likelihood function (1.22), which in turn is dominated by solving the $n(k + 1) \times n(k + 1)$ linear systems at (1.23) and (1.24) that involve the matrix C_θ. This matrix is larger in dimension by a factor of $k + 1$ than it would be if derivatives were not included, so the cost of each likelihood evaluation is increased by a factor of about $(k + 1)^3$. This is clearly a large *relative* increase when the number of inputs is large, but it may still not be of much consequence when compared with the computation needed to run the model code.

Of course, the speed of the computer and the efficiency of the optimization algorithm are also relevant to the consideration of cost involved in using available derivatives. Alternatives to full optimization—for example, the "line search" method proposed by Welch et al. (1992)—may be considered too as a way to reduce this cost. The problems here are similar to those encountered in the standard setting, without derivatives, when one has many observations.

A somewhat different question arises when one does not already have the derivatives in hand but has the *capability* for producing them as well as function values: Then should one always use this capability? Now the cost of producing the derivatives must be considered. When the adjoint method for the model has been already "hand-coded," then y_0 and all of its first partial derivatives can be generated at about twice the cost of evaluating y_0 alone. [Hall et al. (1982) reported a factor of $1\frac{1}{2}$ in their application of the adjoint method to the calculation of over 300 derivatives in a radiative-convective climate model.]

In its current state of development, automatic differentiation is much less attractive for our purposes, although the potential for generating y_0 and all of its first derivatives in roughly a constant multiple M of the time needed to generate y_0 alone has inspired an active research effort (Griewank 1991a, b). In practical implementations, typical values of M currently lie between 10 and 20, although for some models M is less than 5 (Brian Worley, personal communication). It is expected that this factor will be reduced as research in this area progresses. For our purposes, automatic differentiation is likely to be cost-effective only if the dimension k of the region of interest is considerably larger than M. Otherwise, it would be better to spend the computing time generating y_0 at M sites rather than y_0 and its derivatives at a single site. If M for the borehole model were 9, for example, the computational cost for evaluating y_0 and its derivatives at 10 sites using automatic differentiation would be the same as that for evaluating y_0 alone at 90 different sites. One would expect the latter approach to produce better predictions. Indeed, when we did an experiment in which y_0 alone was generated at the sites of a 90-run Latin hypercube, the errors were roughly one-third of those for the best design shown in Figures 3 and 4.

The cost of building a derivative-generating capability for a model code that does not already have one can be quite considerable. It is not unusual for

the hand-adjointing of a complicated code to require more than a person-year of effort, especially if that person is not familiar with the development of the model. Nevertheless, this kind of activity is quite widespread, especially for "generic" codes that will be used many times in many places. The capability for automatic differentiation can be achieved much more quickly, owing to the existence of appropriate software (e.g., Oblow 1985).

Although the design and analysis procedures described here are straightforward in principle, some questions will require further attention. In particular, the type of correlation function used in an analysis may have considerable influence on the predictions. A referee has pointed out that the correlation function (1.16) that we have used here is sometimes criticized because it is excessively *smooth*, in the sense that it corresponds to an infinitely differentiable process. This prodded us into repeating part of the example of Section 4, using instead

$$R_j(s_j - t_j) = 1 - \frac{3(1 - \rho_j)}{2}(s_j - t_j)^2 + \frac{(1 - \rho_j)}{2}|s_j - t_j|^3, \qquad .01 \le \rho_j \le .99,$$

which is a special case of the cubic correlation given in (2.10) of Currin et al. (1991), reparameterized in terms of $\rho_j = R_j(1)$. This correlation corresponds to a process that has only one derivative. For the data generated by the modified maximin design, the maximum likelihood values of $\rho_1, \rho_2, \ldots, \rho_8$ were (.25, .99, .99, .88, .99, .89, .83, .96), and the prediction errors were roughly three times the size of those shown in Figures 3 and 4. The smoother correlation function definitely seems better for this surface and this design. The question of how one can use the data to choose from among types of correlation functions is one which needs to be investigated further. One recourse is to choose the one that produces the highest likelihood. In this example, at least, we found that the smoother correlation would have been selected by this criterion.

6. SUMMARY AND CONCLUSION

We have described an implementation of Bayesian prediction, based on stochastic process priors, for developing an approximation to the output function y_0 of a computer model, using evaluations of y_0 and its first partial derivatives at a set of design sites. The mechanics, which are fairly straightforward, involve the invocation of the appropriate formulas for the covariances among values and derivatives of y_0 at the same and distinct sites, some bookkeeping, and the extra computing required to account for the k additional pieces of information (the derivatives) at each design site. This provides an effective method of synthesizing the information obtained in a sensitivity analysis, which usually comes in the form of derivatives obtained at one or more design sites.

For design, the approach that attempts to maximize the reduction in entropy (for y_0 and its derivatives at unobserved sites) when the intersite correlations are extremely weak, leads to the same kind of design (maximin) that is *optimal* when no derivatives are observed. In the example calculation described here, a comparison of predictions based on a Latin hypercube design and a maximin design lead to mixed results, with the Latin hypercube performing better on the interior of the input domain and the maximin design performing better at the extremes of the region. Two compromise designs, which are constructed in an effort to preserve the strength of both the Latin hypercube structure and the maximin criterion, are more generally successful. We believe that such classes of designs deserve further study, not only for application to experiments where derivatives are available, but also when values of y_0 only are observed.

ACKNOWLEDGMENTS

We thank Brian Worley of the Engineering Physics and Mathematics Division, Oak Ridge National Laboratory, for information on automatic and not-so-automatic methods for calculating derivatives in computer models and for encouragement in using these derivatives for prediction of output functions. Max Morris's and Toby Mitchell's research was sponsored by the Applied Mathematical Sciences Research Program, Office of Energy Research, U.S. Department of Energy, under Contract DE-AC05-84OR21400 with the Martin Marietta Energy Systems, Inc. Donald Ylvisaker's research was supported in part by National Science Foundation Grant DMS 89-02494.

APPENDIX: PROOF THAT ASYMPTOTICALLY *D*-OPTIMAL DESIGNS ARE MAXIMIN DESIGNS

Following the approach of Section 3, we seek to maximize $|\Sigma|$, the determinant of the variance–covariance matrix of $\vec{Y}$ at (1.9), where $K_{00}(t, s) = \text{cov}\,[Y_0(t), Y_0(s)]$ has the form

$$K_{00}(t,s) = \sigma^2 R^\theta(s - t) = \sigma^2 \prod_{j=1}^{k} R_j^\theta(s_j - t_j), \quad \text{(A.1)}$$

$(s - t) = (s_1 - t_1, \ldots, s_k - t_k)$, and each R_j is the correlation function for a differentiable Gaussian process on the real line. We shall assume that each R_j is positive and decreasing. The role of the exponent θ is to control the strength of the intersite correlations. Here we consider the question of design

as these correlations become asymptotically weak; that is, $\theta \to \infty$.

The elements of Σ are determined by applying the standard rule (1.15) for determining covariances among function values and derivative values. Since σ^2 appears as a multiplicative constant in every element of Σ, we shall take $\sigma^2 = 1$ with no loss of generality. Letting $K_{pq}(t, s) = \text{cov}[Y_p(t), Y_q(s)]$ and $g_p(x) = \ln R_p(x)$, we can write

$$K_{00}(t, s) = R^\theta(s - t), \tag{A.2a}$$

$$K_{p0}(t,s) = -\theta R^\theta(s - t)g_p'(s_p - t_p), \quad p \geq 1, \tag{A.2b}$$

$$K_{0q}(t,s) = \theta R^\theta(s - t)g_q'(s_q - t_q), \quad q \geq 1, \tag{A.2c}$$

$$K_{pq}(t, s) = -\theta^2 R^\theta(s - t)g_p'(s_p - t_p)g_q'(s_q - t_q), \quad p \geq 1, q \geq 1, p \neq q, \tag{A.2d}$$

and

$$K_{pp}(t, s) = -\theta^2 R^\theta(s - t)[g_p'(s_p - t_p)]^2 - \theta R^\theta(s - t)g_p''(s_p - t_p), \quad p \geq 1. \tag{A.2e}$$

Simplifications occur when $t = s$, since $R(0, \ldots, 0) = 1$ and $g_p'(0) = 0$. In this case, (A.2e) provides the variance of Y_p:

$$\text{var}[Y_p(t)] = \theta V_p, \tag{A.3}$$

where

$$V_p = -g_p''(0). \tag{A.4}$$

Since these variances are independent of design, maximizing the determinant of Ψ, the *correlation* matrix of $\vec{Y}$, is equivalent to maximizing $|\Sigma|$.

Let Ψ^{ts} be the block of Ψ associated with the generic design sites t and s. The structure of Ψ^{ts} can be discerned from (A.2); Ψ^{tt} is the identity. For $t \neq s$, the upper left corner of Ψ^{ts} is $R^\theta(s - t)$, the remainder of the first row contains terms of the form $\sqrt{\theta}R^\theta(s - t)a_q(t_q, s_q)$, and the remainder of the first column contains terms of the form $-\sqrt{\theta}R^\theta(s - t)a_p(t_p, s_p)$, where

$$a_p(t_p, s_p) = \frac{1}{\sqrt{V_p}} g_p'(s_p - t_p). \tag{A.5}$$

The other diagonal elements of Ψ^{ts}, $s \neq t$, have the form

$$-\theta R^\theta(s - t)a_p^2(t_p, s_p) - \frac{R^\theta(s - t)}{V_p} g_p''(s_p - t_p) \approx -\theta R^\theta(s - t)a_p^2(t_p, s_p), \tag{A.6}$$

the approximation holding for large θ, and the off-diagonal elements have the form

$$-\theta R^\theta(s - t)a_p(t_p, s_p)a_q(t_q, s_q). \tag{A.7}$$

Let $\bar{R}_D$ be the maximum of $R(s - t)$ over all pairs of design sites and let P_D be the set of pairs at which this maximum is attained. Diagonal expansion of $|\Psi|$ (Aitken 1956, p. 87), which we now write as $|\Psi_D|$ to indicate dependence on the design, yields

$$\begin{aligned}|\Psi_D| &= 1 - \sum_{P_D} \theta^2 \bar{R}_D^{2\theta} \sum_p \sum_q a_p^2(t_p, s_p)a_q^2(t_q, s_q) + o(\theta^2 \bar{R}_D^{2\theta}) \\ &= 1 - \theta^2 \bar{R}_D^{2\theta} \sum_{P_D} \sum_p \sum_q a_p^2(t_p, s_p)a_q^2(t_q, s_q) + o(\theta^2 \bar{R}_D^{2\theta}) \\ &= 1 - \theta^2 \bar{R}_D^{2\theta} \sum_{P_D} \left(\sum_p a_p^2(t_p, s_p) \right)^2 + o(\theta^2 \bar{R}_D^{2\theta}).\end{aligned} \tag{A.8}$$

It is evident that a necessary condition for a design to be asymptotically D optimal is that $\bar{R}_D$ be minimized. This result is applicable whenever the correlation structure has the product form.

Consider now the particular case in which $R_p(s_p - t_p) = e^{-(s_p - t_p)^2}$, so $R(s - t) = e^{-d^2(t,s)}$, where $d(t, s)$ is the Euclidean distance

$$d(t, s) = (\Sigma (s_p - t_p)^2)^{1/2}. \tag{A.9}$$

This distance achieves its minimum (over all pairs of design sites) at precisely those pairs that are in P_D. Denote the number of these pairs by $J(D)$ and the corresponding distance by $\underline{d}(D)$. Then $\bar{R}_D = e^{-\underline{d}^2(D)}$. Differentiation of $g_p(x) = -x^2$ and Equations (A.4), (A.5), and (A.9) yield

$$\sum_p a_p^2(t, s) = 2d^2(t, s),$$

which is constant and equal to $2\underline{d}^2(D)$ on P_D, so (A.8) can be written

$$|\Psi_D| = 1 - 4\theta^2 e^{-2\theta \underline{d}^2(D)} J(D)\, \underline{d}^4(D) + o(\theta^2 e^{-2\theta \underline{d}^2(D)}). \tag{A.10}$$

Examination of (A.10) shows that, when the covariance function is $K_{00}(t, s) = \sigma^2 e^{-\theta \Sigma (s_p - t_p)^2}$, a design is asymptotically D optimal as $\theta \to \infty$ only if it maximizes $\underline{d}$ (the minimum intersite Euclidean distance). Among such designs, moreover, only those for which $J(D)$ is minimized can be optimal.

[*Received October 1990. Revised June 1992.*]

REFERENCES

Aitken, A. C. (1956), *Determinants and Matrices* (9th ed.), Edinburgh and London: Oliver and Boyd.

Alsmiller, R. G., Barish, J., Drischler, J. D., Horwedel, J. E., Lucius, J. L., and McAdoo, J. W. (1983), "Sensitivity Theory and Its Application to a Large Energy-Economics Model," *Operations Research*, 31, 915–937.

Bohachevsky, I. O., Johnson, M. E., and Stein, M. L. (1986), "Generalized Simulated Annealing for Function Optimization," *Technometrics*, 28, 209–217.

Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991), "Bayesian Prediction of Deterministic Functions, With Applications to the Design and Analysis of Computer Experiments," *Journal of the American Statistical Association*, 86, 953–963.

Griewank, A. (1989), "On Automatic Differentiation," in *Mathematical Programming: Recent Developments and Applications*, eds. M. Ira and K. Tanabe, Boston: Kluwer, pp. 83–108.

——— (1991a), "The Chain Rule Revisited in Scientific Computing," *SIAM News*, 24, 20–21.

——— (1991b), "The Chain Rule Revisited in Scientific Computing, Part II," *SIAM News*, 24, 8–9, 24.

Hall, M. C. G., Cacuci, D., and Schlesinger, M. E. (1982), "Sensitivity Analysis of a Radiative-Convective Model by the Adjoint Method," *Journal of the Atmospheric Sciences*, 39, 2038–2050.

Harper, W. V., and Gupta, S. K. (1983), "Sensitivity/Uncertainty Analysis of a Borehole Scenario Comparing Latin Hypercube Sampling and Deterministic Sensitivity Approaches," BMI/ONWI-516, Office of Nuclear Waste Isolation, Battelle Memorial Institute, Columbus, OH.

Johnson, M., Moore, L., and Ylvisaker, D. (1990), "Minimax and Maximin Distance Designs," *Journal of Statistical Planning and Inference*, 26, 131–148.

Kirkpatrick, S., Gelatt, C. D., Jr., and Vechhi, M. P. (1983), "Optimization by Simulated Annealing," *Science*, 220, 671–680.

Marable, J. H., Weisbin, C. R., and de Saussure, G. (1980), "Uncertainty in the Breeding Ratio of a Large Liquid-Metal Fast Breeder Reactor: Theory and Results," *Nuclear Science and Engineering*, 75, 30–55.

McKay, M. D., Conover, W. J., and Beckman, R. 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.

Mitchell, T. J., Morris, M. D., and Ylvisaker, D. (1990), "Existence of Smoothed Stationary Processes on an Interval," *Stochastic Processes and their Application*, 35, 109–119.

Oblow, E. M. (1985), "GRESS, Gradient-Enhanced Software System—Version D User's Guide," ORNL/TM-9658, Oak Ridge National Laboratory, Oak Ridge, TN.

Oblow, E. M., Pin, F. G., and Wright, R. Q. (1986), "Sensitivity Analysis Using Computer Calculus: A Nuclear Waste Application," *Nuclear Science and Engineering*, 94, 46.

Owen, A. (1990), "Controlling Correlations in Latin Hypercube Sampling," technical report, available from author.

Park, J.-S. (1991), "Tuning Complex Computer Codes to Data and Optimal Designs," unpublished Ph.D. thesis, University of Illinois, Dept. of Statistics.

Parzen, E. (1962), *Stochastic Processes*, San Francisco: Holden-Day.

Paterson, H. D. (1954), "The Errors of Lattice Sampling," *Journal of the Royal Statistical Society*, Ser. B, 16, 140–149.

Sacks, J., Schiller, S. B., and Welch, W. J. (1989), "Designs for Computer Experiments," *Technometrics*, 31, 41–47.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), "Design and Analysis of Computer Experiments," *Statistical Science*, 4, 409–423.

Shewry, M. C., and Wynn, H. P. (1987), "Maximum Entropy Sampling," *Journal of Applied Statistics*, 14, 165–170.

Stein, M. L. (1989), Discussion of "Design and Analysis of Computer Experiments," by J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, *Statistical Science*, 4, 432–433.

Tang, B. (in press), "OA-Based Latin Hypercubes," *Journal of the American Statistical Association*, 88.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J., and Morris, M. D. (1992), "Screening, Predicting, and Computer Experiments," *Technometrics*, 34, 15–25.

Worley, B. A. (1987), "Deterministic Uncertainty Analysis," ORNL-6428, available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

Worley, B. A., Wright, R. Q., Pin, F. G., and Harper, W. V. (1986), "Application of an Automated Procedure for Adding a Comprehensive Sensitivity Calculation Capability to the ORIGEN2 Point Depletion and Radioactivity Decay Code," *Nuclear Science and Engineering*, 94, 180.

Ylvisaker, D. (1987), "Prediction and Design," *The Annals of Statistics*, 15, 1–19.