Kriging Metamodeling in Simulation

Kleijnen, J.P.C.

Publication date:
2007

CentER

Discussion Paper

No. 2007–13

KRIGING METAMODELING IN SIMULATION: A REVIEW

By Jack P.C. Kleijnen

February 2007

ISSN 0924-7815

Tilburg University

Kriging metamodeling in simulation: a review

Jack P.C. Kleijnen

Department of Information Systems and Management, Tilburg University, Postbox 90153, 5000 LE Tilburg, Netherlands

Abstract

This article reviews Kriging (also called spatial correlation modeling). It presents the basic Kriging assumptions and formulas—contrasting Kriging and classic linear regression metamodels. Furthermore, it extends Kriging to random simulation, and discusses bootstrapping to estimate the variance of the Kriging predictor. Besides classic one-shot statistical designs such as Latin Hypercube Sampling, it reviews sequentialized and customized designs. It ends with topics for future research.

Key words: Kriging, Metamodel, Response Surface, Interpolation, Design
JEL: C0, C1, C9

1 Introduction

Metamodels are also called response surfaces, surrogates, emulators, auxiliary models, etc. By definition, a metamodel is an approximation of the Input/Output (I/O) function that is defined by the underlying simulation model. This simulation model may be either deterministic or random (stochastic). Examples of deterministic simulation are models of airplanes, automobiles, TV sets, and computer chips—applied in Computer Aided Engineering (CAE) and Computer Aided Design (CAD) at Boeing, General Motors, Philips, etc. These deterministic simulations may show numerical inaccuracies, which make this type of simulation related to random simulation. The latter type, however, uses Pseudo-Random Numbers (PRNs) inside its model. Examples are models of logistic and telecommunication systems. In this article, I cover both types of simulation. (Because simulation is applied in so many different disciplines, the terminology varies widely; I try to include different terms for the same concept.)

Most publications on metamodels focus on *low-order polynomial regression*. Such metamodels are fitted to the I/O data of the local or global experiment

with the underlying simulation model. This type of metamodel may be used for the explanation of the underlying simulation model's behavior, and for prediction of the expected simulation output for combinations of input values that have not yet been simulated (inputs are also called factors; combinations are also called scenarios). The final goals of the metamodel may be validation of the simulation model, sensitivity analysis, and optimization.

In this article, I focus on *Kriging* metamodels. Typically, Kriging models are fitted to data that are obtained for larger experimental areas than the areas used in low-order polynomial regression metamodels; that is, Kriging models are *global* rather than local. These models are used for prediction; the final goals are sensitivity analysis and optimization.

Kriging was originally developed in *geostatistics* (also known as spatial statistics) by the South African mining engineer called Danie Krige. The mathematics were further developed by Matheron; see his 1963 article [19]. A classic geostatistics textbook is Cressie's 1993 book [3]. I also recommend References 17 through 21 in [18] .

Later on, Kriging models were applied to the I/O data of *deterministic simulation* models. These models have k-dimensional input where k is a given positive integer (whereas geostatistics considers only two-dimensional input); see Sacks et al.'s classic 1989 article [21]. More recent publications are Jones et al.'s 1998 summary article [8], Simpson et al.'s 2001 article [23], and Santner et al.'s 2003 textbook [22].

Only in 2003, Van Beers and I started applying Kriging to *random simulation* models; see [25]. Although Kriging in random simulation is still rare, I strongly believe that the track record Kriging achieved in deterministic simulation holds promise for Kriging in random simulation!

Note: Searching for 'Kriging' via Google on February 15, 2007 gave 631,000 hits, which illustrates the popularity of this mathematical method. Searching for 'Operations Research' within these pages gave 81,000 hits.

The goal of this article is to review the basics of Kriging, and some recent extensions. I expect that these basics will convince analysts in deterministic or random simulation of the potential usefulness of Kriging. Furthermore, I hope that the review of recent extensions is also of interest to analysts who are already familiar with Kriging in simulation.

The rest of this article is organized as follows. Section 2 covers the basic assumptions and formulas of Kriging. Section 3 presents some relatively new results, including random simulation and estimating the variance of the Kriging predictor through bootstrapping. Section 4 includes one-shot and sequential statistical designs for simulation experiments analyzed through Kriging

metamodels. Section 5 presents conclusions and topics for future research

2 Kriging basics

I start with highlighting the differences between (familiar) linear regression—especially low-order polynomial regression—and Kriging models. I focus on a single (univariate, scalar) simulation output, because most Kriging models also assume such output. My general black-box representation is then

$$w = s(d_1, \ldots, d_k, \mathbf{r}_0) \tag{1}$$

where

w is the output of the underlying simulation model;

$s(.)$ denotes the mathematical function implicitly defined by the computer code implementing this simulation model;

d_j with $j = 1, \ldots k$ is the j^{th} input variable (factor) of the simulation program, so $\mathbf{D} = (d_{ij})$ is the design matrix for the simulation experiment, with $i = 1, \ldots, n$ and n the number of input combinations in that experiment,

$\mathbf{r}_0$ is the vector of PRN seeds.

I point out that $\mathbf{D}$ determines the original simulation input variables (say) z and the corresponding standardized (coded, scaled) regression variables x defined below (3). The design matrix $\mathbf{D}$ is usually standardized; for example, a two-level (fractional) factorial has elements that are either -1 or $+1$; also see [9].

The *first-order polynomial* regression metamodel for (1) is

$$y_{reg} = \beta_0 + \beta_1 d_1 + \ldots \beta_k d_k + e_{reg} \tag{2}$$

where

y_{reg} is the regression predictor of the simulation output w in (1); the subscript reg distinguishes this metamodel from the Kriging metamodel presented below;

$\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_k)'$ is the vector with the parameters of this metamodel;

e_{reg} is the error (residual, noise)—which includes both lack of fit of the metamodel and intrinsic noise caused by the PRNs.

The model in (2) is a special case of the general *linear regression* model

$$\mathbf{y}_{reg} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e}_{reg} \tag{3}$$

where

$\mathbf{y}_{reg}$ denotes the n-dimensional vector with the regression predictor where n is the number of simulated combinations;

$\mathbf{X} = (\mathbf{x}_{ij})$ denotes the $n \times q$ matrix of explanatory regression variables with $\mathbf{x}_{ij}$ the value of variable j in run i ($i = 1, \ldots, n; j = 1, \ldots, q$) (for example, in (2) $q = 1 + k$ including the dummy variable or constant $x_{i0} = 1$ corresponding with β_0);

$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_q)'$ denotes the q-dimensional vector of regression parameters (if there is a dummy variable, then β_1 denotes the intercept in the general regression model, whereas the symbol β_0 denoted the intercept in the specific regression model (2));

$\mathbf{e}_{reg}$ is the vector of residuals in the n factor combinations.

Above, I pointed out that $\mathbf{D}$ determines the standardized regression variables x. Indeed, the first-order polynomial model (2) implies $\mathbf{X} = (\mathbf{1}, \mathbf{D})$ where $\mathbf{1} = (1, \ldots, 1)'$ is an n-dimensional vector with each element being the number one.

The *Least Squares* (LS) estimator (say) $\widehat{\boldsymbol{\beta}}$ of the regression parameter vector $\boldsymbol{\beta}$ in the linear regression model (3) can be derived to be

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{w} \tag{4}$$

where $\mathbf{w} = (w_1, \ldots, w_n)'$ is the n-dimensional vector with 'the' output of the simulation model with input $\mathbf{D}$; 'the' output of combination i is the average output of a constant number of replications, $m_i = m$:

$$\overline{w}_i = \frac{\sum_{r=1}^{m} w_{ir}}{m}. \tag{5}$$

Obviously, in deterministic simulation $m = 1$. If the number of replicates is not constant, then a slightly more complicated formulation is required; see [9].

Hence, the regression predictor for a simulation input (say) $\mathbf{d} = (d_1, \ldots, d_k)'$ is

$$\widehat{\mathbf{y}}_{reg}(\mathbf{d}) = \mathbf{x}(\mathbf{d})'\hat{\boldsymbol{\beta}} = \mathbf{x}(\mathbf{d})'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{w} \tag{6}$$

where the vector of explanatory variables $\mathbf{x}$ is determined by the vector of simulation inputs $\mathbf{d}$; for example, the first-order polynomial model (2) implies $\mathbf{x}(\mathbf{d}) = (1, d_1, \ldots, d_k)'$. The input $\mathbf{d}$ may be a new or an old combination (the old combination is one of the rows in $\mathbf{D}$).

In this article, I focus on the simplest type of Kriging called *Ordinary Kriging*, which assumes

$$w(\mathbf{d}) = \mu + \delta(\mathbf{d}) \tag{7}$$

where

μ is the simulation output averaged over the 'experimental area' (also see the comment below);

$\delta(\mathbf{d})$ is the additive noise that forms a 'stationary covariance process' with zero mean (this process is defined below).

Zeigler et al. [27] call the experimental area the *experimental frame*. I would also call it the domain of admissible scenarios—given the goals of the simulation study (various goals are discussed in [10] and [13]).

By definition, the time series w_t is a *stationary covariance process* if it has a constant mean (say) $E(w_t) = \mu$, a constant variance $var(w_t) = \sigma^2$, and covariances depending only on the *lag* $|t - t'|$; that is, $cov(w_t, w_{t'}) = \sigma_{|t-t'|}$.

The metamodel (7) with its constant μ does not imply a *flat* response surface; see [21]. Instead of this constant μ, *Universal* Kriging uses a regression model. However, Ordinary Kriging often suffices in practice; see [1], [17], [18], and [21].

I point out that Kriging is also used—quite successfully—in deterministic simulation. At first sight it seems strange that the *random* (meta)model (7) can be applied to a *deterministic* simulation model. My interpretation is that the deviations of the simulation output w from its mean μ form a random process—with the characteristics of a 'stationary covariance process' (with zero mean); see δ in (7).

Ordinary Kriging (from now on, briefly called Kriging) uses the following *linear* predictor:

$$y(\mathbf{d}) = \boldsymbol{\lambda}(\mathbf{d}, \mathbf{D})'\mathbf{w}(\mathbf{D}) = \boldsymbol{\lambda}'\mathbf{w} \tag{8}$$

where the weights $\boldsymbol{\lambda}(\mathbf{d}, \mathbf{D})$—abbreviated to $\boldsymbol{\lambda}$—are not constants (whereas $\boldsymbol{\beta}$ in equation 3 is) but decrease with the *distance* between the input $\mathbf{d}$ to be predicted and the 'old' points $\mathbf{D}$; this $\mathbf{D}$ determines the simulation output vector $\mathbf{w}$, so the explicit notation is $\mathbf{w}(\mathbf{D})$ and the simpler notation is $\mathbf{w}$.

To select the *optimal* values for the weights $\boldsymbol{\lambda}$ in (8), a criterion must be selected. In linear regression, the Sum of Squared Residuals is the criterion—which gives the LS estimator (4). Kriging selects the *Best Linear Unbiased Predictor (BLUP)*, which (by definition) minimizes the Mean Squared Errors (MSE) of the predictor:

$$\min_{\boldsymbol{\lambda}} MSE[y(\mathbf{d})] = \min_{\boldsymbol{\lambda}} [E(y(\mathbf{d}) - w(\mathbf{d}))]^2 \tag{9}$$

where $\mathbf{d}$ may be any point (input combination) in the experimental area. Moreover, this minimization must account for the condition that the predictor in *unbiased*:

$$E(y(\mathbf{d})) = E(w(\mathbf{d})). \tag{10}$$

Obviously, in deterministic simulation $E(w(\mathbf{d}))$ may be replaced by $w(\mathbf{d})$. It can be proven that the solution of the constrained minimization problem defined by (9) and (10) implies that the weights of the linear predictor (8) must satisfy the following condition:

$$\sum_{i=1}^{n} \lambda_i = 1 \tag{11}$$

or (in matrix notation) $\mathbf{1}'\boldsymbol{\lambda} = 1$.

Furthermore, the *optimal* weights can be proven to have the values

$$\boldsymbol{\lambda}_o = \boldsymbol{\Gamma}^{-1}[\boldsymbol{\gamma} + \mathbf{1}\frac{1 - \mathbf{1}'\boldsymbol{\Gamma}^{-1}\boldsymbol{\gamma}}{\mathbf{1}'\boldsymbol{\Gamma}^{-1}\mathbf{1}}] \tag{12}$$

where

$\boldsymbol{\Gamma} = (cov(w_i, w_{i'}))$ with $i, i' = 1, \ldots, n$ is the $n \times n$ symmetric and positive semi-definite matrix with the covariances between the 'old' outputs (i.e., outputs of input combinations that have already been simulated);

$\boldsymbol{\gamma} = (cov(w_i, w_0))$ is the n-dimensional vector with the covariances between the n 'old' outputs w_i and w_0, the output of the combination to be predicted (which may be 'new' or 'old').

Finally, it can be proven (also see [14]) that (7), (8), and (12) imply

$$y = \widehat{\mu} + \boldsymbol{\gamma}'\boldsymbol{\Gamma}^{-1}(\mathbf{w} - \widehat{\mu}\mathbf{1}) \tag{13}$$

with

$$\widehat{\mu} = (\mathbf{1}'\mathbf{R}^{-1}\mathbf{1})^{-1}\mathbf{1}'\mathbf{R}^{-1}\mathbf{w}.$$

It is easy to see that the Kriging model in (7) implies $E(y) = \mu$ because (7) implies $E(\mathbf{w}) = \mu\mathbf{1}$. Furthermore, if (say) $w_1 > \mu, w_2 = \mu, \ldots, w_n = \mu$, then the conditional expected value of the predictor in (13) exceeds the unconditional mean μ because $\boldsymbol{\gamma}'\boldsymbol{\Gamma}^{-1} > \mathbf{0}'$.

Obviously, the optimal values for the Kriging weights in (12) depend on the covariances—or equivalently the correlations—between the simulation outputs in the Kriging model (7). Kriging assumes that these correlations are determined by the 'distance' between the inputs of the outputs w_i and $w_{i'}$ or w_i and w_0—or (more succinctly) between w_i and w_g with $g = 0, 1, \ldots, n$.

In simulation applications of Kriging, the usual assumption is that the correlation function for a k-dimensional input vector is the *product* of k one-dimensional functions:

$$\rho(w(\mathbf{d}_i), w(\mathbf{d}_g)) = \prod_{j=1}^{k} \rho(d_{ij}, d_{gj}). \tag{14}$$

Moreover, Kriging assumes a *stationary covariance process*, which implies that the correlations depend only on

$$h_j(i,g) = |d_{ij} - d_{gj}| \ \ (j = 1, \ldots, k) \ (i = 1, \ldots, n)(g = 0, 1 \ldots, n). \tag{15}$$

So, $\rho(d_{ij}, d_{gj})$ in (14) reduces to $\rho(h_j(i,g))$. Transforming the standardized design points d_j into the original simulation inputs z_j makes the distances scale dependent; also see [2].

There are several types of stationary covariance processes. Three popular types for a single input (so $h_j = h$ in equation 15) with parameter $\theta > 0$ are:

- Linear correlation function: $\rho(h) = \max(1 - \theta h, 0)$
- Exponential correlation function: $\rho(h) = \exp(-\theta h)$
- Gaussian correlation function: $\rho(h) = \exp(-\theta h^2)$ (its point of inflection can be proven to be $1/\sqrt{2\theta}$).

In Kriging, a popular correlation function is

$$\rho(\mathbf{h}) = \exp[-\sum_{j=1}^{k} (\frac{h_j}{\theta_j})^{p_j}] = \prod_{j=1}^{k} \exp[-(\frac{h_j}{\theta_j})^{p_j}] \tag{16}$$

where

θ_j denotes the importance of factor j; that is, the higher θ_j is, the less effect input j has;

p_j denotes the smoothness of the correlation function; for example, $p_j = 2$ implies an infinitely differentiable function. Obviously, exponential and Gaussian correlation functions have $p = 1$ and $p = 2$ respectively.

Correlation functions that decrease as the distance increases, imply that the optimal weights are relatively high for inputs close to the input to be predicted. Furthermore, some of the weights may be *negative*. Finally, the weights imply that for an 'old' input (so $\mathbf{d}$ is a row within $\mathbf{D}$) the predictor equals the observed simulation output at that input:

$$y(\mathbf{d}_i) = w(\mathbf{d}_i) \text{ if } \mathbf{d}_i \in \mathbf{D}, \tag{17}$$

so all weights are zero except the weight of the observed output. This property implies that the Kriging predictor is an exact *interpolator*, whereas the regression predictor minimizes the Sum of Squared Residuals (SSR) so it is not an exact interpolator—unless $n = q$ (where q was defined below equation 3)

A major problem is that the optimal Kriging weights λ_i depend on the correlation function of the underlying simulation model—*but this correlation function is unknown*. Therefore both the type and its parameter values must be estimated. (The number of observations for a covariance of a given distance h decreases as that distance increases.) Given these estimates for various values of the distance h, a correlation function is fitted. To estimate the parameters of such a correlation function, the standard software and literature uses Maximum Likelihood Estimators (MLEs). A MLE requires constrained maximization. This optimization is a hard problem, because matrix inversion is necessary, multiple local maxima may exist, etc.; see [16] and [18]. (Besides the MLE criterion, [18] uses cross-validation; for the linear correlation function, [11] uses the LS criterion.)

For the estimation of the correlation functions and the optimal weights through (12), I recommend the Matlab Kriging toolbox DACE, which is free of charge; see [15]. Alternative free software is available via http://www.stat.ohio-state.edu/~comp_exp/ and http://endo.sandia.gov/Surfpack.

Note: There are also many publications that interpret Kriging models in a *Bayesian* way; see [7] and [16].

3 Kriging: new results

The interpolation property in (17) is attractive in *deterministic* simulation, because the observed simulation output is unambiguous (ignoring numerical noise that may occur when deterministic simulation software is executed; see [24]). In *random* simulation, however, the observed output is only one of the many possible values. For random simulations, [25] replaces $w(\mathbf{d}_i)$ in (9) by the average observed output

$$\overline{w}_i = \frac{\sum_{r=1}^{m_i} w_{ir}}{m_i} \; (i = 1, \ldots, n). \tag{18}$$

These n averages, however, are still random, so the property in (17) loses its intuitive appeal. Nevertheless, [25] gives examples in which the Kriging predictions based on (18) are much better than the regression predictions (regression metamodels may be useful for other goals such as understanding, screening,

and validation). (Reference [22] includes a computer program in C—called PErK—which allows random output, but I do not know any applications.)

The Kriging model in (7) assumes a stationary covariance process, which implies a constant variance (say) σ_δ^2. However, in experiments with random simulation models such as queueing models, the output variances $var(w_i)$ are not constant at all! Fortunately, [12] demonstrates that the Kriging model is not very sensitive to this variance heterogeneity.

The following problem is virtually ignored in the Kriging literature: replacing the weights in (8) by the estimated optimal weights (say) $\widehat{\boldsymbol{\lambda}_0}$ implies that the Kriging predictor becomes a *nonlinear* estimator. The literature uses the predictor variance—*given* the Kriging weights $\boldsymbol{\lambda}$. At a fixed point $\mathbf{d}$, this variance follows directly from (12) (also see [3], p. 122):

$$var[y(\mathbf{d})|\boldsymbol{\lambda})] = 2\sum_{i=1}^{n} \lambda_i cov(w_0, w_i) - \sum_{i=1}^{n}\sum_{i'=1}^{n} \lambda_i \lambda_{i'} cov(w_i, w_{i'}). \qquad (19)$$

Using (19), it is easy to derive that the variance in case w_0 equals w_i reduces to zero.. Ignoring the randomness of the estimated optimal weights tends to underestimate the true variance of the Kriging predictor Moreover, the true (unconditional) variance and the conditional variance do not reach their maxima for the same input combination. To estimate the true variance, I first discuss random simulation; then deterministic simulation.

In *random* simulation, each input combination is replicated a number of times; also see (18) Therefore a simple method for estimating the true predictor variance is *distribution-free bootstrapping*. The basics of bootstrapping are explained in [5] and [9]. To estimate the predictor variance in Kriging, [26] resamples—with replacement—the m_i Independent and Identically Distributed (IID) observations. This sampling results in the bootstrapped average $\overline{w}_i^*$ where the superscript $*$ is the usual symbol to denote a bootstrapped observation and $i = 1, \ldots, n$. From these n bootstrapped averages $\overline{w}_i^*$, the bootstrapped estimated optimal weights $\widehat{\boldsymbol{\lambda}_0}^*$ and the corresponding bootstrapped Kriging predictor y^* are computed. To decrease sampling effects, this whole procedure is repeated B times, which gives y_b^* with $b = 1, \ldots, B$. The variance of the Kriging predictor is estimated from these B values.

For *deterministic* simulation, [4] applies *parametric* bootstrapping, assuming a Gaussian stationary covariance process and Gaussian correlation functions; see (16) with $p = 2$. The 'true' parameter values θ_j of this process are estimated from the given simulation I/O data, ($\mathbf{D}$, $\mathbf{w}$). Next, the Monte Carlo method is used to sample bootstrapped data ($\mathbf{D}, \mathbf{w}^*$) from the estimated multivariate normal distribution. For alternative approaches (including cross-validation and Akaike's Information Criterion) I refer to [16] and [18].

4 Designs for Kriging

Simulation analysts often use *Latin Hypercube Sampling* (LHS) to generate the I/O simulation data to which they fit a Kriging (meta)model. Actually, LHS was not invented for Kriging but for Risk Analysis; see [9].

LHS assumes that an adequate metamodel is more complicated than a low-order polynomial such as (2), which is assumed by classic designs such as fractional factorials. LHS, however, does not assume a specific metamodel or simulation model. Instead, LHS focuses on the design space formed by the k–dimensional unit cube defined by the standardized simulation inputs. LHS is one of the space filling types of design. Other designs related to LHS (e.g., maximin designs and orthogonal arrays) are discussed in [9],including many references and websites.

An alternative for LHS are *sequentialized* designs. Sequential statistical procedures are known to be more 'efficient'; that is, they require fewer observations than fixed-sample (one-shot) procedures; see, for example, [20]. Moreover, computer experiments (unlike agricultural experiments) proceed sequentially. Nevertheless, sequential procedures may be less efficient computationally; for example, re-estimating the Kriging parameters may be costly; see [6].

In [11] and [26], Van Beers and I develop a sequential procedure for deterministic and random simulations respectively. These two procedures share the following steps.

(1) We start with a *pilot* experiment, using some small space-filling design (e.g., a LHS design). Its (say) n_0 combinations form the input into the simulation model, and results in the corresponding simulation outputs.
(2) We fit a *Kriging* model to the I/O simulation data that is available at this step (in the first pass of this procedure, these I/O data are the data resulting from Step 1).
(3) We consider (but do not yet simulate) a set of *candidate* input combinations that have not yet been simulated and that are selected through some space-filling design; we select as the next combination to be actually simulated, the candidate combination that has the highest *predictor variance* (below, I discuss how to estimate this variance)
(4) We use the combination selected in Step 3 as the input to the simulation model, run the (expensive) simulation, and obtain the corresponding simulation output.
(5) We re-fit a Kriging model to the I/O data that is augmented with the I/O data resulting from Step 4.
(6) We return to Step 3 until we are satisfied with the Kriging metamodel.

Our designs are also *customized* (tailored or application-driven, not generic);

that is, which combination has the highest predictor variance is determined by the underlying simulation model. For example, if the simulation model has an I/O function (response surface) that is a simple hyperplane within a subspace of the total experimental area, then our procedure selects relatively few points in that part of the input space.

We experiment with two *random* simulation models that are building blocks for more realistic simulation models. The first model is an M/M/1 simulation with a traffic rate d that varies over the experimental area $0.1 \leq d \leq 0.9$. First we use a LHS design with $n = 10$ prefixed values for the traffic rate. Next we use our procedure, until we have simulated the same number of observations as in the LHS design (namely 10). For each simulated traffic rate, we obtain enough IID observations (namely, renewal cycles) to reach a prespecified accuracy (namely, a confidence interval with a prespecified type-I error rate, say, α and a prespecified relative error); also see [13]. We apply distribution-free bootstrapping to these IID observations, to estimate the Kriging predictor variance (see Step 3 in our procedure presented above). Our design turns out to select more input values in the part of the input range that gives a drastically increasing (highly nonlinear) I/O function. Our design gives better Kriging predictions than the fixed LHS design does—especially for small designs, which are used in expensive simulations. The second model is an (s, S) inventory simulation. Again, in the 'rough' region of the response surface our procedure simulates more input combinations.

We also experiment with a *deterministic* simulation model, namely a fourth-degree polynomial I/O function in one input variable with two local maxima and three local minima; two of these minima are at the border of the experimental area. Now we do not use bootstrapping to estimate the predictor variance in Step 3 of our procedure; instead, we use cross-validation and jackknifing (as explained below). We compare our design to a sequential design based on (19), which approximates the variance of the Kriging predictor ignoring the random character of the estimated weights. The latter design selects as the next point the input value that maximizes this variance; i.e., there is no need to specify candidate points. It turns out that this approach selects as the next point the input farthest away from the old inputs, so the final design spreads all its points evenly across the experimental area—like space filling designs do.

We, however, estimate the true predictor variance through *cross-validation*. So, we successively delete one of the I/O observations already simulated from the original I/O data set $(\mathbf{D}, \mathbf{w})$, which gives $(\mathbf{D}_{-i}, \mathbf{w}_{-i})$ where the subscript $-i$ means that $(\mathbf{d}_i, w_i)$ is deleted. Next, we recompute the Kriging prediction, based on the recomputed correlation function parameters and the corresponding optimal Kriging weights.

To quantify this prediction uncertainty, we use *jackknifing*. So, we calculate the jackknife's pseudovalue (say) J for candidate input combination (say) j as the weighted average of the original and the cross-validation predictors:

$$J_{j;i} = n\widehat{y_j} - (n-1)\widehat{y_{j;-i}} \text{ with } j = 1, \ldots, c \text{ and } i = 1, \ldots, n$$

where c denotes the number of candidate points and n the number of points that have really been simulated so far and are deleted successively. From these pseudovalues we compute the classic variance estimator:

$$\widehat{var}(J_j) = \frac{\sum_{i=1}^{n}(J_{j;i} - \overline{J_j})^2}{(n-1)}.$$

Like in the two random simulation examples, our design favors input combinations in subareas that have more interesting I/O behavior; i.e., our final design selects relative few input values in the subareas that generate an approximately linear I/O function, whereas it selects many input values near the borders, where the function changes much.

In [9], I briefly review several publications that also propose sequentialized and customized designs for simulation. Those publications use a Bayesian approach, the conditional variance formula in (19), other metamodels than Kriging (e.g., splines), etc.

5 Conclusions and future research

This article may be summarized as follows.

- I started with a review of the basic assumption of Kriging, namely 'old' simulation observations closer to the new point to be predicted, should receive more weight. This assumption is formalized through a stationary covariance process with correlations that decrease as the distances between observations increase.
- Moreover, the Kriging model is an interpolator; i.e., predicted outputs equal observed simulated outputs at old points.
- Next, I reviewed some more recent results for random simulation, and I explained how the true variance of the Kriging predictor can be estimated through bootstrapping.
- I finished with a discussion of one-shot and sequentialized, customized designs for simulation experiments to be analyzed through Kriging.

There is a need for more research:

- Kriging *software* needs further improvement. For example, the estimated metamodel should be less sensitive to the prespecified lower and upper limits for the correlation parameters θ_j. The usual MSE criterion may be replaced by the maximum squared error criterion. In random simulation, Kriging still assumes that the predictors equal the average outputs at the inputs already observed.
- Sequentialized and customized design procedures may benefit from *asymptotic proofs* of their performance; for example, does the design approximate the optimal design (the latter designs are briefly discussed in [9])?
- More experimentation and analyses may be done to derive *rules of thumb* for our sequential procedure's parameters, such as the size of the pilot design in deterministic or random simulation and the initial number of replicates in random simulation experiments. For this pilot design and for the set of candidate points not only LHS but also other space-filling designs may be studied; for example, maximin designs and orthogonal arrays.
- *Stopping rules* for sequential designs based on a measure of accuracy (or precision) may be investigated.
- Kriging should also be applied to *practical* random simulation models, which are more complicated than the academic M/M/1 queueing and (s, S) inventory models.
- Whereas I focused on Sensitivity Analysis, other researchers search for the *optimal* input of the simulation model. Can both approaches be combined, especially in robust optimization (also see [9])?
- Nearly all Kriging publications assume univariate output, whereas in practice simulation models have *multivariate output*.
- Often the amalysts know that the simulation's I/O function has certain properties, for example, monotonicity. Most metamodels (Kriging, regression) do not preserve these properties (also see [9]).
- Sequential and customized designs may be analyzed not only through Kriging but also through other types of metamodels.

Acknowledgements

I thank Dick den Hertog and Wim van Beers (both at Tilburg University) for their comments on an earlier version of this article.

References

[1] Chen, V.C.P., K.-L. Tsui, R.R. Barton, and J.K. Allen (2003), A review of design and modeling in computer experiments. In: *Handbook of Statistics;*

Volume 22, edited by R. Khattree and C.R. Rao, Elsevier, Amsterdam, pp. 231-261

[2] Crary, S.B., D.M. Woodcock, and A. Hieke (2001), Designing efficient computer experiments for metamodel generation. *Proceedings Modeling and Simulation of Microsystems Conference*, pp. 132-135

[3] Cressie, N.A.C. (1993), *Statistics for spatial data: revised edition.* Wiley, New York

[4] Den Hertog, D., J.P.C. Kleijnen, and A.Y.D. Siem (2006), The correct Kriging variance estimated by bootstrapping. *Journal of the Operational Research Society*, 57, no. 4, pp. 400-409

[5] Efron, B. and R.J. Tibshirani (1993), *An introduction to the bootstrap.* Chapman & Hall, New York

[6] Gano, S. E., J. E. Renaud, J.D. Martin, and T.W. Simpson (2006), Update strategies for Kriging models for using in variable fidelity optimization, *Structural and Multidisciplinary Optimization*, 32, no. 4, pp. 287-298

[7] Hankin, R.K.S. (2005), Introducing BACCO, an R bundle for Bayesian analysis of computer code output. *Journal of Statistical Software*, 14, no. 16, pp. 1-21

[8] Jones, D.R., M. Schonlau, W.J. Welch (1998), Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13, pp. 455-492

[9] Kleijnen, J.P.C. (2007), *DASE: Design and analysis of simulation experiments.* Springer Science + Business Media

[10] Kleijnen, J.P.C. and R.G. Sargent (2000). A methodology for the fitting and validation of metamodels in simulation. *European Journal of Operational Research*, 120, no. 1, pp. 14-29

[11] Kleijnen, J.P.C. and W.C.M. van Beers (2004), Application-driven sequential designs for simulation experiments: Kriging metamodeling. *Journal of the Operational Research Society,* 55, no. 9, pp. 876-883

[12] Kleijnen, J.P.C. and W.C.M. van Beers (2005), Robustness of Kriging when interpolating in random simulation with heterogeneous variances: some experiments . *European Journal of Operational Research*, 165, no. 3, pp. 826-834

[13] Law, A.M. and W.D. Kelton (2000), *Simulation modeling and analysis; third edition.* McGraw-Hill, Boston

[14] Lin, Y., F. Mistree, J.K. Allen, K-L Tsui, V. Chen, Sequential Metamodeling in Engineering Design, 10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Albany, NY, August 30-September 1, 2004, Albany, NY. Paper Number AIAA-2004-4304

[15] Lophaven, S.N., H.B. Nielsen, and J. Sondergaard (2002), DACE: a Matlab Kriging toolbox, version 2.0. IMM Technical University of Denmark, Lyngby

[16] Martin, J.D., and T.W. Simpson (2004), A Monte Carlo simulation of the Kriging model. *10th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA-2004-4483*

[17] Martin, J.D., and T.W. Simpson (2004), On using Kriging models as probabilistic models in design. *2004-01-0430*

[18] Martin, J.D., and T.W. Simpson (2005), Use of Kriging models to approximate deterministic computer models. *AIAA Journal*, 43, no. 4, pp. 853-863

[19] Matheron, G. (1963), Principles of geostatistics. *Economic Geology*, 58, no. 8, pp. 1246-1266

[20] Park, S., J.W. Fowler, G.T. Mackulak, J.B. Keats, and W.M. Carlyle (2002), D-optimal sequential experiments for generating a simulation-based cycle time-throughput curve. *Operations Research*, 50, no. 6, pp. 981-990

[21] Sacks, J., W.J. Welch, T.J. Mitchell and H.P. Wynn (1989), Design and analysis of computer experiments (includes Comments and Rejoinder). *Statistical Science*, 4, no. 4, pp. 409-435

[22] Santner, T.J., B.J. Williams, and W.I. Notz (2003), *The design and analysis of computer experiments*. Springer-Verlag, New York

[23] Simpson, T.W., T.M. Mauery, J.J. Korte, and F. Mistree (2001), Kriging metamodels for global approximation in simulation-based multidisciplinary design optimization. *AIAA Journal*, 39, no. 12, pp. 2233-2241

[24] Toropov, V.V., U. Schramm, A. Sahai, R. Jones, and T. Zeguer (2005), Design optimization and stochastic analysis based on the Moving Least Squares method. *6th World Congress of Structural and Multidisciplinary Optimization*, Rio de Janeiro, paper no. 9412

[25] Van Beers, W. and J.P.C. Kleijnen (2003), Kriging for interpolation in random simulation. *Journal of the Operational Research Society*, no. 54, pp. 255-262

[26] Van Beers, W.C.M. and J.P.C. Kleijnen (2006), Customized sequential designs for random simulation experiments: Kriging metamodeling and bootstrapping. Working Paper, Tilburg University, Tilburg, Netherlands

[27] Zeigler B.P., H. Praehofer, T.G. Kim (2000), *Theory of modeling and simulation; second edition.* Academic Press, San Diego