

Tilburg University

Kriging in Multi-response Simulation, including a Monte Carlo Laboratory (Replaced by 2014-012)

Kleijnen, Jack P.C.; Mehdad, E.

Publication date:
2012

[Link to publication in Tilburg University Research Portal](#)

Citation for published version (APA):

Kleijnen, J. P. C., & Mehdad, E. (2012). *Kriging in Multi-response Simulation, including a Monte Carlo Laboratory (Replaced by 2014-012)*. (CentER Discussion Paper; Vol. 2012-039). Operations research.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



No. 2012-039

KRIGING IN MULTI-RESPONSE SIMULATION,
INCLUDING A MONTE CARLO LABORATORY

By

Jack Kleijnen, Ehsan Mehdad

May 16, 2012

ISSN 0924-7815

Kriging in Multi-response Simulation, including a Monte Carlo Laboratory

Jack P.C. Kleijnen, Ehsan Mehdad

CentER, TiSEM, Tilburg University, Postbox 90153, 5000 LE Tilburg, Netherlands,
{kleijnen@tilburgvt.nl, E.Mehdad@uvt.nl}

To analyze the input/output behavior of simulation models with multiple responses, we may apply either univariate or multivariate Kriging (Gaussian Process) models. Univariate Kriging may use a popular MATLAB Kriging toolbox called “DACE”. Multivariate Kriging faces a major problem: its covariance matrix should remain positive-definite; this problem may be solved through nonseparable dependence model. To evaluate the performance of these two Kriging models, we develop a Monte Carlo “laboratory” that simulates Gaussian Processes. To verify that this laboratory works correctly, we derive statistics that test whether the Kriging parameters have the correct values. Our Monte Carlo results demonstrate that in general DACE gives smaller Mean Squared Error (MSE); we also explain these results.

Key words: positive-definite covariance-matrix, nonseparable dependence model, Gaussian process, verification

JEL: C0, C1, C9, C15, C44

1. Introduction

In practice, *simulation* is applied in many scientific disciplines—ranging from sociology to astronomy, as described in the excellent survey by Karplus (1983). Simulation may be either *deterministic* or *random* (stochastic). Applications of deterministic simulation abound in engineering (e.g., in Computer Aided Engineering), but there are also applications in Operations Research / Management Science (OR/MS). An application is the management of fisheries at the French Research Institute for Exploitation of the Sea (IFREMER); see Mahevas and Pelletier (2004). Another application is the case study on the CO_2 greenhouse effect by Kleijnen et al. (1992). Applications of random simulation are plentiful in OR/MS (e.g., in queueing and inventory management); see the many references in Kleijnen (2008, pp. 3-6).

Kriging may be used to analyze the Input/Output (I/O) behavior of a given simulation model (this analysis may serve validation, sensitivity analysis, and optimization; see Kleijnen

et al. (2010). This Kriging gives a *metamodel*, which approximates the I/O function defined by the underlying simulation model (metamodels are also called response surfaces, surrogates, emulators, etc. in different disciplines). There are different types of metamodels; the most popular type is a polynomial of first or second order; see Kleijnen (2008, pp. 15-100). We, however, focus on Kriging, which has already become popular in engineering (see again Kleijnen (2008)) and is gaining popularity in OR/MS (see the references in Chen et al. (2010)).

In practice, a given simulation model has *multiple outputs* (responses, performance criteria). For example, one case study concerned the production planning of steel tubes of different types, using a simulation model with 28 outputs which—after a discussion with management—were reduced to two outputs; for details see Kleijnen (1993). Multiple performance metrics in supply chain management are discussed by Kleijnen and Smits (2003). The literature on metamodels, however, often reduces these multiple outputs to a single output—either ignoring all the other outputs or combining all outputs through a weighting function. We will examine two examples; namely, the sum and the product of two outputs. Other publications present a metamodel for each *individual* output ignoring the correlations between outputs; e.g., Kleijnen et al. (2010) fit univariate Kriging models for each of the two outputs (cost and service) of a call-center simulation. We will examine examples of such univariate Kriging.

To empirically investigate univariate and multivariate Kriging, we use *Monte Carlo* experiments that control the validity (adequacy) of the Kriging model. The literature usually experiments with realistic simulation models, but these experiments mix-up the approximation errors of the Kriging metamodels and the errors of the metamodel variants. We limit this investigation to Kriging in deterministic simulation, which is also the basis for Kriging in stochastic simulation. Our Monte Carlo examples require relatively little computer time; in practice, however, simulation may be computationally expensive so Kriging is attractive.

There are rather few publications that explicitly discuss multiple outputs. Cressie (1991, pp. 138-142) speaks of *cokriging* in his book on spatial data analysis. Wackernagle (2003, pp. 143-209) also discusses geostatistics, so he restricts the input data to one, two, or three dimensions (whereas simulation implies an arbitrary number of dimensions). He allows more complicated correlation functions than we and other simulation researchers do; e.g., Wackernagle (2003, pp. 62-65) allows so-called “anisotropic” correlation functions. Gneiting et al. (2010) also discuss cokriging in geostatistics assuming so-called Matérn correlation functions.

Santner et al. (2003, pp. 101-116) do discuss simulation or computer experiments, but they do not use the same assumptions as we do; e.g., they discuss a simulation model that gives a single output plus its first-order derivative. Higdon et al. (2008) discuss the combination of real-life “field data” and simulation data, where both types of data concern the same real-life system; they allow for very many types of output. Forrester (2010) also discusses the combination of (i) scarce and expensive real-life data with abundant and inexpensive simulation data, or (ii) scarce and expensive data from a detailed simulation model with abundant and inexpensive data from a quick-and-dirty simulation model. Williams et al. (2010) discuss multivariate Kriging in constrained optimization in simulation with multiple outputs—but they follow Santner et al. (2003). Altogether we recommend Santner et al. (2003) and Wackernagle (2003) for an introduction to multivariate Kriging. Note that Li et al. (2006) also recognize that in practice simulation models have multiple outputs and that Kriging is an important type of metamodel, but those authors use a completely different approach (they do not use cokriging with estimated cross-correlations). Chan et al. (2009) discuss multi-fidelity modeling, which is also discussed in multivariate Kriging.

In this article, we summarize a multivariate Kriging model with nonseparable covariance structure proposed by Fricker et al. (2010). Furthermore, we try to interpret this novel Kriging model. We also present empirical comparisons of the performance of this multivariate model and univariate Kriging per output. These results demonstrate that multivariate Kriging does not outperform univariate Kriging.

The remainder of this article is organized as follows. Section 2 summarizes the basics of univariate Kriging. Section 3 extends this Kriging to multivariate Kriging with nonseparable dependence structure. Section 4 describes our Monte Carlo laboratory with Gaussian Process (GP) models so the Kriging assumptions are controlled; we propose several statistical tests to verify the correctness of this lab, and use this laboratory to numerically compare univariate and multivariate Kriging. Section 5 presents conclusions and topics for future research. The many references at the end of this article enable the reader to study more aspects of this challenging topic.

2. Basic univariate Kriging

Table 5 in Appendix A lists our major symbols; our notation remains close to the notation in DACE—the free univariate MATLAB Kriging toolbox developed and well-documented by

Lophaven et al. (2002b), where they assume deterministic simulation. Note that alternative free software is mentioned by Frazier (2010) and Kleijnen (2008, p. 146); commercial software called JMP is offered by SAS. Several authors present a Bayesian interpretation of the Kriging model, but we follow a frequentist approach.

Suppose the given simulation model is run for m combinations of the k simulation inputs $\mathbf{x} = (x_1, \dots, x_k)^T$. (We may call these combinations “locations” or “scenarios”; Lophaven et al. (2002b) call them “sites”, which stems from the origin of Kriging; namely, geostatistics; Danie Krige was a mining engineer in South Africa). Simulating these m input combinations gives the outputs $\mathbf{y} = (y_1, \dots, y_m)^T$. (Lophaven et al. (2002b, p. 1) define a multi-variate output, but their software is limited to univariate output.)

Like most Kriging publications in simulation, we assume *Ordinary Kriging* (whereas DACE allows Universal Kriging, which uses a linear regression model instead of our constant μ ; Universal Kriging would use the symbol β_0 instead of μ):

$$y = \mu + z \tag{1}$$

where μ denotes the mean output, and z a *stationary Gaussian process* with zero mean. Because this process is stationary, it has a constant (but unknown) variance σ_z^2 , and the covariances $c_{i,i'}$ ($i, i' = 1, \dots, m$) between the outputs of the input combinations \mathbf{x}_i and $\mathbf{x}_{i'}$ are determined by the *distance* between these \mathbf{x}_i and $\mathbf{x}_{i'}$ in the k -dimensional input space (we use the symbol $c_{i,i'}$ instead of $\sigma_{i,i'}$ because in multivariate Kriging $\sigma_{g,g'}$ refers to the covariances between the outputs of type g and g'). Kriging of simulation models with their possibly high-dimensional input space assumes that these covariances are the *products* of the k individual correlation functions; e.g., a so-called *Gaussian correlation function* implies

$$c_{i,i'} = \sigma_z^2 \prod_{j=1}^k \exp[-\theta_j (x_{j,i} - x_{j,i'})^2] \tag{2}$$

where $\theta_j \geq 0$ measures the importance of input j and $|x_{j,i} - x_{j,i'}|$ measures the distance between the i and i' input combination of input j . Note that if $\theta_j = 0$, then changes in input j have no effect at all on the covariance $c_{i,i'}$. If $\theta_j = \infty$, then the covariance $c_{i,i'}$ reduces to zero, so the outputs at the locations i and i' are independent. The covariances $c_{i,i'}$ are gathered in the symmetric and positive-definite $m \times m$ covariance matrix $\mathbf{\Sigma}$, and the corresponding correlations in \mathbf{R} so $\mathbf{\Sigma} = \sigma_z^2 \mathbf{R}$. The two extreme values for the correlation coefficient ($\theta_j = 0$ or $\theta_j = \infty$) give a singular covariance matrix, because this matrix has

identical columns. Note that Simpson et al. (2001) state that Ordinary Kriging with a Gaussian correlation function is the most common Kriging model in engineering.

The classic Kriging predictor assumes *known (hyper)parameters* μ and Σ . (den Hertog et al. (2006) use bootstrapping to account for the estimation of these parameters). Requiring the predictor to be unbiased and assuming the Mean Squared Prediction Error (MSPE) criterion, Lophaven et al. (2002b, pp. 3-4) prove that the *optimal Kriging predictor* for the output y_0 of \mathbf{x}_0 (the “new” input combination at point 0; an alternative notation replaces the subscript 0 by $m + 1$) is

$$\hat{y}_0 = \mu + \mathbf{c}_0^T \Sigma^{-1} (\mathbf{y} - \mu \mathbf{1}) \quad (3)$$

where $\mathbf{1}$ denotes the m -dimensional vector with ones; $\mathbf{c}_0 = (c_{0;1}, \dots, c_{0;m})^T$ the vector with the covariances between the outputs at the new and the m old input combinations (so $\Sigma = (\mathbf{c}_1, \dots, \mathbf{c}_m)$ with vectors \mathbf{c}_i); $(\mathbf{y} - \mu \mathbf{1})$ the vector with residuals. (If the new point \mathbf{x}_0 is actually one of the old points \mathbf{x}_i with $i = 1, \dots, m$, then the predictor \hat{y}_i equals the observed output y_i so Kriging gives an exact interpolator.)

The mean squared prediction error (MSPE) or the estimated variance of the Kriging predictor (3) is

$$\sigma_z^2 + (\mathbf{1}^T \Sigma^{-1} \mathbf{c}_0)^T (\mathbf{1}^T \Sigma^{-1} \mathbf{1})^{-1} (\mathbf{1}^T \Sigma^{-1} \mathbf{c}_0) - \mathbf{c}_0^T \Sigma^{-1} \mathbf{c}_0 \quad (4)$$

In practice, however, these parameters are unknown, so a major problem is the *estimation* of μ and Σ (which includes the variance σ_z^2 on the main diagonal and the k parameters θ_j if we assume the Gaussian correlation function (2)). Like nearly all authors, Lophaven et al. (2002b) use *Maximum Likelihood Estimation* (MLE). Because z in (1) follows a *Gaussian process*, the distribution function (say) f of \mathbf{y} (the m -dimensional vector with outputs) is

$$f(\mathbf{y}) = \frac{1}{(2\pi)^{m/2} (|\Sigma|)^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{y} - \mu \mathbf{1})^T \Sigma^{-1} (\mathbf{y} - \mu \mathbf{1}) \right] \quad (5)$$

where $|\Sigma|$ denotes the determinant of Σ ; a similar formula is given by Santner et al. (2003, p. 210). This function (5) may be denoted by $\mathcal{N}_m(\boldsymbol{\mu}, \Sigma)$ with $\boldsymbol{\mu} = \mu \mathbf{1}$.

Ignoring terms that do not depend on the parameters to be estimated (μ and Σ), (5) obviously implies that the *log-likelihood function* to be minimized is

$$l(\Sigma, \mu | \mathbf{y}) = \ln |\Sigma| + (\mathbf{y} - \mu \mathbf{1})^T \Sigma^{-1} (\mathbf{y} - \mu \mathbf{1}). \quad (6)$$

Minimizing this $l(\Sigma, \mu | \mathbf{y})$ gives the MLE estimators (say) $\hat{\Sigma}$ and $\hat{\mu}$. This minimization is a difficult mathematical problem. The classic solution in Kriging is to “divide and conquer” through the application of mathematical statistics, as follows.

We have already noticed the relationship $\Sigma = \sigma_z^2 \mathbf{R}$. So in (6) we replace $|\Sigma|$ by $|\mathbf{R}| (\sigma_z^2)^m$ (an elementary property of determinants) and Σ^{-1} by $\mathbf{R}^{-1}/\sigma_z^2$:

$$l(\mathbf{R}, \mu | \mathbf{y}) = [m \ln \sigma_z^2 + \ln |\mathbf{R}|] + \frac{(\mathbf{y} - \mu \mathbf{1})^T \mathbf{R}^{-1} (\mathbf{y} - \mu \mathbf{1})}{\sigma_z^2}. \quad (7)$$

Following Lophaven et al. (2002b) and also Gano et al. (2006), we minimize this function in the following steps:

1. Initialize; i.e., select preliminary values for $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \dots, \hat{\theta}_k)^T$ which together define $\hat{\mathbf{R}}$.
2. Compute the Generalized Least Squares (GLS) estimator of the mean:

$$\hat{\mu} = (\mathbf{1}^T \hat{\mathbf{R}}^{-1} \mathbf{1})^{-1} \mathbf{1}^T \hat{\mathbf{R}}^{-1} \mathbf{y}. \quad (8)$$

3. Substitute $\hat{\mu}$ resulting from Step 2 and $\hat{\mathbf{R}}$ resulting from Step 1 into the estimate of the variance

$$\hat{\sigma}_z^2 = \frac{(\mathbf{y} - \hat{\mu} \mathbf{1})^T \hat{\mathbf{R}}^{-1} (\mathbf{y} - \hat{\mu} \mathbf{1})}{m}. \quad (9)$$

Note that this MLE estimator uses the denominator m , whereas the classic unbiased estimator assuming $\mathbf{R} = \mathbf{I}$ would use $m - 1$. Furthermore, Antognini and Zagoraiou (2010, p. 14) state: “under the Ordinary Kriging model in the one-dimensional case the covariance parameters θ and σ^2 cannot be identified simultaneously”.

4. Solve the remaining problem in (7):

$$\underset{\boldsymbol{\theta}}{\text{Min}} \left[m \ln \hat{\sigma}_z^2 + \ln |\hat{\mathbf{R}}| \right]. \quad (10)$$

To solve this nonlinear minimization problem, Lophaven et al. (2002a, p. 29) apply the classic Hooke-Jeeves heuristic. Gano et al. (2006) point out that this minimization problem is difficult because of “the multimodal and long near-optimal ridge properties of the likelihood function”.

5. Use the $\hat{\boldsymbol{\theta}}$ that solves (10) to update $\hat{\mathbf{R}}$, and substitute the resulting $\hat{\mathbf{R}}$ into (8) and (9).

Computational aspects are further discussed by Lophaven et al. (2002b) and Lophaven et al. (2002a), including the classic Cholesky and QR factorization, regularization of an ill-conditioned $\hat{\mathbf{R}}$, and normalization of the I/O such that all data have zero means and unit variances (we shall return to these aspects). Numerical problems in Kriging are also discussed by Jones et al. (1998, p. 486) and Marrel et al. (2010, p. 5). The problem of a flat likelihood function leading to highly variable MLE is tackled by Li and Sudjianto (2005), adding a penalty function to the likelihood function.

3. Multivariate Kriging

In this section we consider $n \geq 1$ outputs for each of the m input combinations; i.e., the simulation outputs become $y_{i;g}$ ($i = 1, \dots, m$) ($g = 1, \dots, n$) (also see Tables 5 and 6 in Appendixes A and B). In multivariate Kriging, we can still use the *multinormal* distribution defined for the univariate case in (5)—provided we define the stacked vector (say) \mathbf{Y} with mn elements such that we first gather the n outputs at the first input combination $\mathbf{y}_1 = (y_{1;1}, \dots, y_{1;n})^T$ (the first row of Table 6), then the n outputs at the second input combination $\mathbf{y}_2 = (y_{2;1}, \dots, y_{2;n})^T$, ..., finally the n outputs at the m^{th} input combination $\mathbf{y}_m = (y_{m;1}, \dots, y_{m;n})^T$ (the last row of Table 6). Note that y_g (output of type g with $g = 1, \dots, n$) has the constant mean μ_g (see Table 5). The resulting vector \mathbf{Y} has the multivariate normal distribution function $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$ where $\boldsymbol{\mu}$ denotes the mean vector with mn elements and $\boldsymbol{\Sigma}_{\mathbf{Y}}$ denotes the $mn \times mn$ covariance matrix of \mathbf{Y} :

$$f(\mathbf{Y}) = \frac{1}{(2\pi)^{mn/2}(|\boldsymbol{\Sigma}_{\mathbf{Y}}|)^{1/2}} \exp \left[-\frac{1}{2}(\mathbf{Y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1}(\mathbf{Y} - \boldsymbol{\mu}) \right]. \quad (11)$$

For $\boldsymbol{\Sigma}_{\mathbf{Y}}$ we offer the following comments.

In Appendix A we detail the simplest multivariate Kriging example; namely, a single input ($k = 1$) and two outputs ($n = 2$). In this section we discuss the general case with k inputs and n outputs. A stationary Gaussian process for each type of output implies that the output y_g has the constant variance σ_g^2 and (auto)covariances that decrease with the distance between its input combinations; again see (2). Moreover, in multivariate Kriging different output types $y_g(\mathbf{x}_i)$ and $y_{g'}(\mathbf{x}_{i'})$ with $g, g' = 1, \dots, n$ and $g \neq g'$ have (cross)covariances, when simulated for the same or for different input combinations; i.e., \mathbf{x}_i may be the same as $\mathbf{x}_{i'}$ or may be different. For example, if $n = 2$ (bivariate output), then

$$Cov(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})) = \begin{bmatrix} cov(y_1(\mathbf{x}_i), y_1(\mathbf{x}_{i'})), & cov(y_1(\mathbf{x}_i), y_2(\mathbf{x}_{i'})) \\ cov(y_1(\mathbf{x}_i), y_2(\mathbf{x}_{i'})), & cov(y_2(\mathbf{x}_i), y_2(\mathbf{x}_{i'})) \end{bmatrix}.$$

If in this example (with $n = 2$) we have $\mathbf{x}_i = \mathbf{x}_{i'}$, then the 2×2 matrix (say) $\boldsymbol{\Sigma}_0$ does not vary with the input combination \mathbf{x} and becomes

$$\boldsymbol{\Sigma}_0 = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} \\ \sigma_{1;2} & \sigma_2^2 \end{bmatrix} \quad (12)$$

with $\sigma_{1;2} = cov(y_1, y_2)$. In the general case with n outputs, the (symmetric) covariance

matrix at input combination i ($i = 1, \dots, m$) is

$$\boldsymbol{\Sigma}_0 = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} & \dots & \sigma_{1;n} \\ & \sigma_2^2 & \dots & \sigma_{2;n} \\ & & \ddots & \vdots \\ & & & \sigma_n^2 \end{bmatrix}. \quad (13)$$

So in the general case \mathbf{Y} has the $mn \times mn$ covariance matrix

$$\boldsymbol{\Sigma}_{\mathbf{Y}} = \begin{bmatrix} \boldsymbol{\Sigma}_0 & Cov(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_2)) & \dots & Cov(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_m)) \\ Cov(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_2)) & \boldsymbol{\Sigma}_0 & \dots & Cov(\mathbf{Y}(\mathbf{x}_2), \mathbf{Y}(\mathbf{x}_m)) \\ \vdots & \vdots & \dots & \vdots \\ Cov(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_m)) & Cov(\mathbf{Y}(\mathbf{x}_2), \mathbf{Y}(\mathbf{x}_m)) & \dots & \boldsymbol{\Sigma}_0 \end{bmatrix}. \quad (14)$$

To *predict* the n outputs at an input combination (say) \mathbf{x}_0 , we define (analogously to \mathbf{c}_0 defined below (3)) the $n \times mn$ matrix

$$\boldsymbol{\Sigma}_{0;m;n} = (Cov(\mathbf{Y}(\mathbf{x}_0), \mathbf{Y}(\mathbf{x}_1)), \dots, Cov(\mathbf{Y}(\mathbf{x}_0), \mathbf{Y}(\mathbf{x}_m)))$$

and obtain the Kriging predictor (analogously to (3))

$$\hat{\mathbf{y}}(\mathbf{x}_0) = \hat{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{0;m;n} \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} (\mathbf{Y} - \mathbf{F} \hat{\boldsymbol{\mu}}), \quad (15)$$

where the vector $\hat{\boldsymbol{\mu}}$ denotes the GLS estimator defined analogously to (8):

$$\hat{\boldsymbol{\mu}} = (\mathbf{F}^T \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} \mathbf{Y} \quad (16)$$

with $\mathbf{F} = \mathbf{1}_m \otimes \mathbf{I}_n$ where $\mathbf{1}_m$ denotes an m -dimensional vector with ones, \otimes is the Kronecker operator, and \mathbf{I}_n denotes the $n \times n$ unity matrix; also see Svenson and Santner (2010).

The MSPE of the multivariate Kriging predictor (15) is

$$\boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_{0;m;n} \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} \boldsymbol{\Sigma}_{0;m;n}^T + \mathbf{U} \times (\mathbf{F}^T \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} \mathbf{F})^{-1} \times \mathbf{U}^T \quad (17)$$

where $\mathbf{U} = \mathbf{I}_n - \boldsymbol{\Sigma}_{0;m;n} \boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} \mathbf{F}$. Note: Wackernagle (2003, pp. 145-150)—also see Gneiting et al. (2010)—warns against wrong specifications of covariance matrices; i.e., $\boldsymbol{\Sigma}_{\mathbf{Y}}$ should be a *Positive Definite* (PD) matrix (so $\mathbf{x}' \boldsymbol{\Sigma}_{\mathbf{Y}} \mathbf{x} > 0$ for all non-zero vectors \mathbf{x} ; there exists a unique lower triangular matrix \mathbf{L} with strictly positive diagonal elements such that $\boldsymbol{\Sigma}_{\mathbf{Y}} = \mathbf{L} \mathbf{L}^T$ —so-called Cholesky decomposition; a PD matrix has a corresponding inverse matrix). Wackernagle (2003, pp. 151-152) (also see Gneiting et al. (2010) for a similar discussion assuming Matérn correlation functions) points out that it is difficult to characterize

a correlation function matrix directly, but Cramér (1940)’s generalization of Bochner’s theorem enables generalization in the *spectral* domain. Using Cramér’s theorem, Yaglom (1987, pp. 315-317) gives the example of a bivariate exponential correlation function that has a PD matrix of spectral densities. We tried this approach, but did not succeed in deriving practical results; e.g., for the case of two outputs ($n = 2$), and one input ($k = 1$), we derive the constraint $\theta^{(1;2)} = \theta^{(1)} = \theta^{(2)}$ (so both outputs behave the same). Yaglom (1987, pp. 315-317) derives a different constraint for exponential cross-correlation functions. Iacoa et al. (2011) give an advanced discussion of PD covariance matrixes in geostatistics. Boyle and Frean (2004) give another way to find a PD covariance matrix using so-called “convolution kernels”. In the next subsection we consider a way to obtain a $\Sigma_{\mathbf{Y}}$ that is indeed PD.

3.1 Nonseparable dependence model

Our discussion of nonseparable dependence model follows Svenson and Santner (2010), who in turn follow Fricker et al. (2010); the latter authors discuss both nonseparable and separable models. Remember that an n -variate Normal variable *without* auto-correlation—denoted by $\mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ —may be generated from a vector \mathbf{Z} with n Normally Independently Identically Distributed (NIID) “standard” variables (so their means are zero and their variances are one) through $\boldsymbol{\mu} + \mathbf{AZ}$ with $\boldsymbol{\Sigma} = \mathbf{AA}^T$ where \mathbf{A} is a symmetric matrix. Svenson and Santner (2010) extend this idea, and consider

$$\mathbf{Y} = \boldsymbol{\mu} + \mathbf{AZ} \tag{18}$$

where \mathbf{Y} denotes the n -variate output at any input combination, $\boldsymbol{\mu}$ is an n -dimensional vector with means μ_g , $\mathbf{A} = (a_{g;g'})$ is a symmetric and PD matrix, and \mathbf{Z} is a vector of mutually independent stationary Gaussian processes with zero mean and unit variance. More precisely, \mathbf{Z} has the Gaussian correlation function defined in (2). It is simple to derive that (18)—together with \mathbf{R} defined below (2)—implies

$$Cov(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})) = \mathbf{A}diag[\mathbf{R}(\mathbf{x}_i - \mathbf{x}_{i'}; \boldsymbol{\theta}^{(1)}), \dots, \mathbf{R}(\mathbf{x}_i - \mathbf{x}_{i'}; \boldsymbol{\theta}^{(n)})]\mathbf{A}^T. \tag{19}$$

When $\mathbf{x}_i = \mathbf{x}_{i'}$ (so (2) gives $\exp[-\theta_j(x_{j;i} - x_{j;i'})^2] = 1$), (19) implies that Σ_0 defined in (13) becomes

$$Cov(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_i)) = \Sigma_0 = \mathbf{AA}^T. \tag{20}$$

Hence, $\sigma_{g;g'}$ (the covariance between y_g and $y_{g'}$) and $\sigma_{g;g} = \sigma_g^2$ (the variance of y_g) are

$$\sigma_{g;g'} = \sum_{l=1}^n a_{g;l} a_{g';l} \quad (g, g' = 1, \dots, n). \quad (21)$$

For example, for $n = 2$ we get (because \mathbf{A} is symmetric, we have $a_{g;g'} = a_{g';g}$)

$$\Sigma_0 = \begin{bmatrix} a_{1;1}^2 + a_{1;2}^2 & a_{1;1}a_{2;1} + a_{1;2}a_{2;2} \\ a_{2;1}a_{1;1} + a_{2;1}a_{2;2} & a_{2;1}^2 + a_{2;2}^2 \end{bmatrix}. \quad (22)$$

Note that each element $a_{g;g'} (= a_{g';g})$ affects the two variances and the covariance (we shall detail this characteristic in the next section). Assuming a Gaussian correlation function (2) and a single input x , (19) becomes $R((x_i - x_{i'}; \theta^{(g)})) = \exp[-\theta^{(g)}(x_i - x_{i'})^2] = \exp[-\theta^{(g)}d_{i;i'}^2]$ with $d_{i;i'} = |x_i - x_{i'}|$ so

$$\text{Cov}[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})] = \mathbf{A} \begin{bmatrix} \mathbf{R}(d_{i;i'}; \theta^{(1)}) & 0 \\ 0 & \mathbf{R}(d_{i;i'}; \theta^{(2)}) \end{bmatrix} \mathbf{A}^T$$

so $\text{Cov}[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})]$ is

$$\begin{bmatrix} a_{1;1}^2 \mathbf{R}(d_{i;i'}; \theta^{(1)}) + a_{1;2}^2 \mathbf{R}(d_{i;i'}; \theta^{(2)}) & a_{1;1} \mathbf{R}(d_{i;i'}; \theta^{(1)}) a_{2;1} + a_{1;2} \mathbf{R}(d_{i;i'}; \theta^{(2)}) a_{2;2} \\ a_{1;1} \mathbf{R}(d_{i;i'}; \theta^{(1)}) a_{2;1} + a_{1;2} \mathbf{R}(d_{i;i'}; \theta^{(2)}) a_{2;2} & a_{2;1}^2 \mathbf{R}(d_{i;i'}; \theta^{(1)}) + a_{2;2}^2 \mathbf{R}(d_{i;i'}; \theta^{(2)}) \end{bmatrix}. \quad (23)$$

Following Fricker et al. (2010) and Svenson and Santner (2010), we select \mathbf{A} as the unique square root of Σ_0 while guaranteeing that \mathbf{A} is PD. We therefore use the Cholesky transformation, $\mathbf{A} = \mathbf{L}\mathbf{L}^T$. We should ensure that all the elements on the main diagonal of \mathbf{L} are non-negative; i.e., we should impose the constraint $l_{i;i} \geq 0$ ($i = 1, \dots, n$) in the MLE optimization.

Actually, Svenson and Santner (2010) apply *Restricted* MLE (RMLE) instead of MLE (for details on RMLE see Santner et al. (2003, pp. 66-67)). The RMLE $\hat{\mathbf{A}}$ (which must be PD) and $\hat{\Theta}$ (which is the multivariate analogue of the correlation coefficients $\hat{\theta}$ defined below (7) in Step 1) minimize the analogue of (7); namely,

$$l(\Sigma_{\mathbf{Y}}, \mu | \mathbf{Y}) = \ln |\Sigma_{\mathbf{Y}}| + \ln |\mathbf{F}^T \Sigma_{\mathbf{Y}}^{-1} \mathbf{F}| + (\mathbf{Y} - \mathbf{F}\hat{\mu})^T \Sigma_{\mathbf{Y}}^{-1} (\mathbf{Y} - \mathbf{F}\hat{\mu}). \quad (24)$$

We call the method defined in this subsection (on NonSeparable dependence model) “NSDep”. In our Monte Carlo experiments (in the next section) we use RMLE for both DACE and NSDep to better compare the two methods. RMLE for DACE requires replacing m by $m - 1$ in (7), (9), and (10).

4. Monte Carlo laboratory using $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$

Why do we need a *laboratory* instead of *real* applications? Kriging is based on specific assumptions; e.g., we assume a GP. To analyze the performance of the resulting Kriging procedure, we should start with situations that satisfy these assumptions; a “laboratory” can fully satisfy the assumptions (Wikipedia states: “A laboratory (lab) is a facility that provides controlled conditions in which scientific research, experiments, and measurement may be performed.”) Real applications enable us to study the “robustness” of the Kriging method; i.e., how well does the method perform if not all its assumptions are completely satisfied? We claim that before we perform such robustness studies, we should examine the performance if all assumptions hold. Moreover, real applications may be extremely expensive; i.e., a single simulation run may take hours or days; in the lab a “simulation” run (an observation) takes only (micro)seconds (depending on the computer hardware and software).

Note that the Kriging literature derives formulas for the estimated variance of the predictor in univariate and multivariate Kriging respectively. These formulas are popular, but we do not use them to compare univariate and multivariate Kriging because we estimate the MSE from the *known* I/O function for the simple systems that we simulate in our lab. Moreover, these formulas are biased because they ignore the variability caused by the estimation of the parameters of the GP; see den Hertog et al. (2006) and Kleijnen et al. (2011).

To compare the performance of multivariate Kriging with univariate Kriging, we use the MSE criterion—after all, this criterion gave the “optimal” Kriging predictor defined in (3). (Other criteria are the closely related Root MSE and the Average Absolute Relative Error; see Kleijnen and Sargent (2000, p.20)) Furthermore, we briefly consider a second criterion; namely, the coverage of the 90% confidence interval for the predictor in univariate versus multivariate Kriging. (Fricker et al. (2010, pp. 13-14) also use criteria closely related to our two criteria.)

We wish to guarantee that the Kriging metamodel itself is a *valid* metamodel (or “adequate” approximation) of the I/O function implied by the underlying simulation model. Therefore we generate the “simulation” observations \mathbf{Y} from $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$ (defined in equation 11) to obtain the I/O data (of Table 6 in Appendix B). To these I/O data we apply univariate and multivariate Kriging respectively, and compare their MSEs and coverages. Note that a similar Monte Carlo lab is used by Chen et al. (2010) for the “empirical evalu-

ation” of their stochastic Kriging.

To specify our lab, we must select the parameter values in $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$. To illustrate our lab, we choose the simplest example; namely, a bivariate output so $n = 2$ and a single input so $k = 1$ (see again Appendix A). We decide to obtain old “simulation” outputs at $m = 10$ input combinations (input values, because there is only $k = 1$ input variable so there are no “combinations”); this choice agrees with the value $10k$ often recommended in the literature; also see the “practical guidelines” in Loepky et al. (2009). Because space-filling designs are most popular in Kriging, we select these m values equi-spaced in the standardized experimental domain $0 \leq x_i \leq 1$ (software like DACE standardizes the k inputs such that each input ranges between 0 and 1; see equations 27 and 28 below), so $\mathbf{x}^T = (0, 1/9, 2/9, \dots, 8/9, 1)^T$. We decide to predict the simulation output for each new input value halfway its two immediate neighbors, so we define the nine-dimensional vector $\mathbf{x}_0^T = (1/18, 3/18, \dots, 17/18)^T$. When we sample from the multivariate normal, we should also sample the “true” simulation output at these new input values \mathbf{x}_0^T ; i.e., in our illustration we sample $(10 + 9 =) 19$ bivariate outputs:

$$\mathbf{Y}_{(38 \times 1)} = \begin{pmatrix} (y_{1;1}, y_{1;2})^T \\ \vdots \\ (y_{19;1}, y_{19;2})^T \end{pmatrix} \sim \mathcal{N}_{(38 \times 1)} [\boldsymbol{\mu}_{(38 \times 1)}, \boldsymbol{\Sigma}_{(38 \times 38)}]. \quad (25)$$

In all our illustrative experiments, we select all ($n = 2$) means equal to zero so in (25) $\boldsymbol{\mu}_{(38 \times 1)} = (0, \dots, 0)^T$. Furthermore, we wish to experiment with “high” and “low” values for the variances σ_1^2 and σ_2^2 , but the problem is that variances—and cross-covariances and auto-covariances—depend on \mathbf{A} in (19) in case of nonseparable dependence model. In our experiments with two outputs, the values we select for the variances and the cross-covariance together with (22) implies

$$\boldsymbol{\Sigma}_0 = \begin{bmatrix} a_{1;1}^2 + a_{1;2}^2 & a_{1;1}a_{2;1} + a_{1;2}a_{2;2} \\ a_{2;1}a_{1;1} + a_{2;1}a_{2;2} & a_{2;1}^2 + a_{2;2}^2 \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} \\ \sigma_{1;2} & \sigma_2^2 \end{bmatrix}, \quad (26)$$

so we have three equations for the three variables $a_{1;1}$, $a_{1;2}$ ($= a_{2;1}$), and $a_{2;2}$. We select $\sigma_1^2 = 1$, $\sigma_2^2 = 25$ (so $\sigma_2 = 5$, which quantifies variability better than its square, σ_2^2), and $\sigma_{1;2} = \rho_{1;2}\sigma_1\sigma_2 = 1$ (so $\rho_{1;2} = 0.2$) and $\sigma_{1;2} = 4$ (so $\rho_{1;2} = 0.8$); i.e., in all our experiments we keep σ_1^2 and σ_2^2 fixed, while we experiment with a low and a high cross-correlation (the cross-correlation remains constant across input combinations). For the auto-covariances we assumed a Gaussian (auto)correlation function and a single input, and derived (23). In (23)

Table 1: Design for Monte Carlo experiment

Correlation	Experiments			
	1	2	3	4
cross: $\rho^{(1;2)}$	0.8	0.8	0.2	0.2
auto: $\rho^{(g)}(d_{min})$	0.8	0.2	0.8	0.2

we have already selected all elements of \mathbf{A} (namely, $a_{1,1}$, $a_{1,2} = a_{2,1}$, and $a_{2,2}$) when selecting the variances and the cross-correlation. To simplify our further selection, we select equal Kriging parameters $\theta^{(1)} = \theta^{(2)} = \theta$ for the two outputs; this changes $Cov[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})]$ in (23) into

$$\begin{bmatrix} (a_{1,1}^2 + a_{1,2}^2)\mathbf{R}(d_{i,i'}; \theta) & (a_{1,1}a_{2,1} + a_{1,2}a_{2,2})\mathbf{R}(d_{i,i'}; \theta) \\ (a_{1,1}a_{2,1} + a_{1,2}a_{2,2})\mathbf{R}(d_{i,i'}; \theta) & (a_{2,1}^2 + a_{2,2}^2)\mathbf{R}(d_{i,i'}; \theta) \end{bmatrix}.$$

We wish to experiment with low and high auto-correlations. The Gaussian auto-correlation function (2) implies that $cor(y_{i;1}, y_{i';1}) = \exp[-\theta(x_i - x_{i'})^2]$. Obviously, these correlations decrease with the distance $d_{i,i'} = |x_i - x_{i'}|$. These distances $d_{i,i'}$ vary with m (number of old equidistant input values in the experimental range) and the $m - 1$ new input values to be predicted (which we selected halfway the old values). Given the standardization of the input range such that $0 \leq x \leq 1$, these distances range between $1/[(m - 1)/2] = 1/18$ (closest neighbors) and 1 (neighbors farthest away). We decide to focus on the strongest auto-correlation between old input values $\rho^1(d_{min}) = \exp[-\theta/9^2]$. For example, $\rho^1(d_{min}) = 0.2$ implies—after rounding— $\theta = 131$ and $\rho^1(d_{min}) = 0.8$ implies $\theta = 18$ (these values are far away from the two extreme values 0 and ∞ , discussed below equation 2). Altogether, Table 1 shows our four experiments combining “low” and “high” values for the cross-correlation $\rho^{(1;2)}$ and the maximum auto-correlation for output g denoted by $\rho^{(g)}(d_{min})$ ($g = 1, 2$).

We decide to repeat this sampling 100 times using non-overlapping (pseudo) random number streams; i.e., we obtain $M = 100$ *macro-replicates*. To *verify* our computer code, we use these macro-replicates to statistically test various intermediate results; namely, the means, variances, and correlations of the simulated bivariate GP output (25).

In general, our Monte Carlo experiments give n outputs $y_{t,g}$ for m old input combinations and $m - 1$ new combinations so $t = 1, \dots, 2m - 1$ and $g = 1, \dots, n$. We use DACE for the univariate Kriging. DACE and NSDep *standardize* the old outputs y_i through the linear

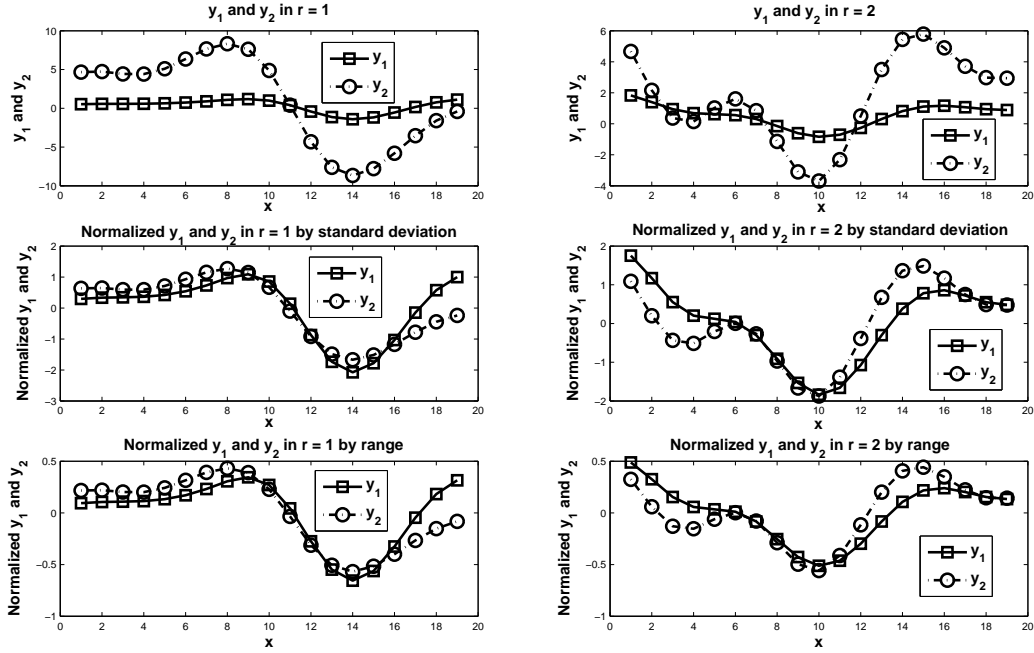


Figure 1: $y_{t;g}$ for macro-replicates 1 and 2 in experiment 1

transformation $(y_i - \bar{y})/s$ where \bar{y} and s are shorthand notations for

$$\bar{y}_g(m) = \frac{\sum_{i=1}^m y_{i;g}}{m} \text{ with } g = 1, \dots, n \quad (27)$$

and

$$s_g(m) = \sqrt{\frac{\sum_{i=1}^m [y_{i;g} - \bar{y}_g(m)]^2}{m}}. \quad (28)$$

Notice that the correlation coefficients are not affected by the linear transformations implied by standardization; however, the Kriging parameters (such as $\theta^{(g)}$) are affected, so we shall present empirical results for non-standardized outputs for both DACE and NSDep.

The Monte Carlo experiments in Table 1 give Figure 1, which displays $y_{t;g}$ with $t = 1, \dots, 19$ and $g = 1, 2$ for macro-replicates 1 and 2 of experiment 1. These plots suggest that the non-standardized outputs have indeed zero means, and that output 2 has higher variability. The three plots also suggest that the outputs are auto-correlated (plots for higher auto-correlation are not displayed). We augment this visual analysis with statistical tests, as follows.

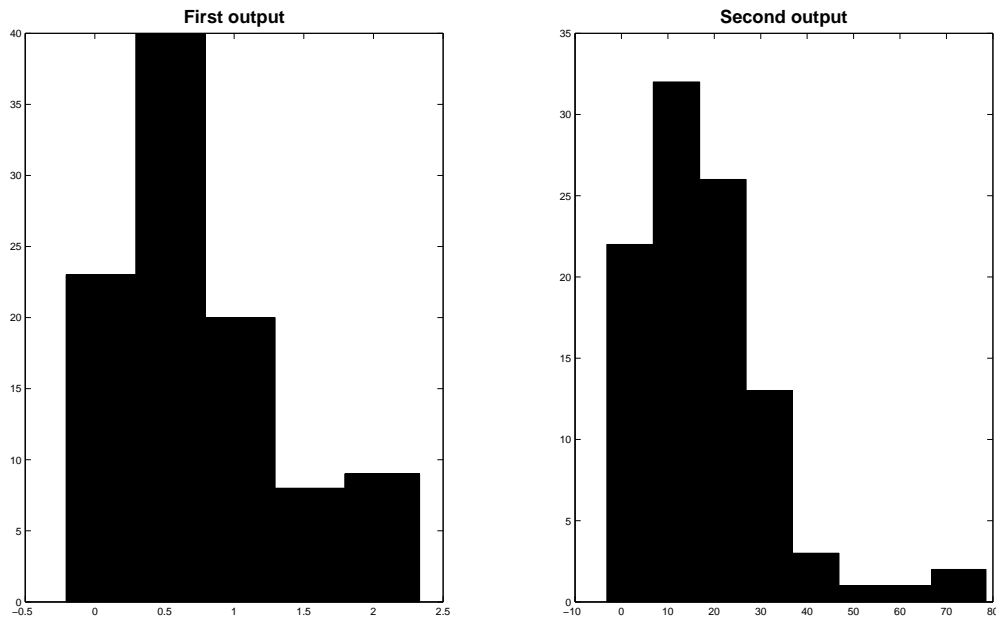


Figure 2: $s_1^2(19)$ and $s_2^2(19)$ in 100 macro-replicates of experiment 1

4.1 Verification of the Monte Carlo laboratory

We derive four tests to verify that our Monte Carlo laboratory has no errors.

(i) We test whether the *averages* $\bar{y}_g(m)$ (see equation 27) are indeed close to the true value $\mu_g = 0$ (see equation 25). Unfortunately, $s_g^2(m)$ defined in (28) is not an unbiased estimator of the variance σ_g^2 , because the outputs at different input combinations are positively correlated; i.e., $s_g^2(m)$ underestimates, as is illustrated by Figure 2 which is a histogram of all M estimates.

Therefore we test our computer code using the macro-replicates, which by definition are Independently Identically Distributed (IID); i.e., defining $y_{t;g;r}$ as output g at input combination t in macro-replicate r gives

$$\bar{y}_{t;g}(M) = \frac{\sum_{r=1}^M y_{t;g;r}}{M} \quad (t = 1, \dots, 2m - 1; g = 1, \dots, n) \quad (29)$$

and the unbiased variance estimators for output g of input combination t

$$s_{t;g}^2(M) = \frac{\sum_{r=1}^M [y_{t;g;r} - \bar{y}_{t;g}(M)]^2}{M - 1}. \quad (30)$$

To test the null-hypothesis that the mean output g is zero, we use the (Student) t statistic

with $M - 1$ Degrees of Freedom (DF):

$$t_{M-1}^{(t;g)} = \frac{\overline{y_{t;g}}(M) - 0}{s_{t;g}(M)/\sqrt{M}}. \quad (31)$$

Because M is high (namely, 100 in our example), we may use the standard Gaussian distribution $N(0, 1)$ for the $t_{M-1}^{(t;g)}$ distribution. In experiment 1 (Table 1) we find that (e.g.) for input value 1 and output 1, (31) gives $t_{100-1}^{(1;1)} = 0.123085/0.1162399 = 1.06$. Altogether we have ($2 \times 19 =$) 38 observations on (31), so we use Bonferroni's inequality; i.e., we replace $\alpha/2$ (two-sided test) by $\alpha/(2 \times 38)$ and we select $\alpha = 0.20$ (this "experimentwise" error rate is higher than the classic 10% or 5%). We find that none of the observed t statistics is significant at any reasonable type-I error probability.

(ii) We test whether the *variance* of output g is indeed σ_g^2 . The GP assumption implies that the variances remain constant at all $2m - 1$ input combinations t (we should not "pool" the $2m - 1$ variance estimators, because they are not independent, as $y_{t;g;r}$ and $y_{t';g;r}$ are correlated). We use χ_{M-1}^2 , which denotes the chi-square statistic with $(M - 1)$ DF. Again using Bonferroni's inequality, we replace $\alpha/2$ by $\alpha/(2 \times 19)$ in the example, and select $\alpha = 0.20$; so $\alpha/38 = 0.005$. In the example, only one of the 19 points in experiment 4 is significant; we decide not to reject our Monte Carlo experiments.

(iii) Analogously to the variance estimator (30) we define the *covariance* estimator

$$s_{t;t'}^{(g;g')}(M) = \frac{\sum_{r=1}^M [y_{t;g;r} - \overline{y_{t;g}}(M)][y_{t';g';r} - \overline{y_{t';g'}}(M)]}{M - 1}, \quad (32)$$

which defines estimators for auto-covariances ($g = g'$) and cross-covariances ($g \neq g'$); obviously (30) is a special case of (32) (to improve the layout, we use $(g;g')$ as a superscript instead of a subscript). Using (32), we obtain the estimated *linear correlation coefficient* $\hat{\rho}$:

$$\hat{\rho}_{t;t'}^{(g;g')}(M) = \frac{s_{t;t'}^{(g;g')}(M)}{s_{t;g}(M)s_{t';g'}(M)}. \quad (33)$$

So the cross-correlation coefficient between outputs g and g' at combination t —estimated from the M macro-replicates—is

$$\hat{\rho}_t^{(g;g')}(M) = \frac{s_t^{(g;g')}(M)}{s_{t;g}(M)s_{t;g'}(M)}. \quad (34)$$

We wish to test whether this coefficient deviates significantly from its expected true value ρ (in the example, this ρ is determined by σ_1^2 , σ_2^2 , and $\sigma_{1;2}$). For this test we use Press et al. (1992, pp. 637-638) to define

$$\frac{z - \omega}{\sigma_z} \text{ with } z = \frac{1}{2} \ln \frac{1 + \hat{\rho}}{1 - \hat{\rho}}, \omega = \frac{1}{2} [\ln \{ \frac{1 + \rho}{1 - \rho} \} + \frac{\rho}{M - 1}], \sigma_z = \frac{1}{\sqrt{M - 3}}. \quad (35)$$

This $(z - \omega)/\sigma_z$ has a $N(0, 1)$ distribution asymptotically. We again use Bonferroni's inequality. In our example, we find that none of the estimated cross-correlations differs significantly from the known values $\rho^{(1;2)}$ in Table 1.

Because (34) is a ratio estimator, we know that this estimator is *biased*. We also know that *jackknifing* reduces the bias of such an estimator; see the overview including references in Kleijnen (2008, pp. 81-84). Jackknifing is a simple statistical technique, which in this case works as follows. First we compute the so-called pseudo-value, which is a weighted combination of the original estimator $\hat{\rho}_t^{(g;g')}(M)$ —abbreviated to $\hat{\rho}_t^{(g;g')}$ —and the estimator deleting replication r denoted by $\hat{\rho}_{t;-r}^{(g;g')}(M-1)$ or briefly $\hat{\rho}_{t;-r}^{(g;g')}$:

$$J_r = M\hat{\rho}_t^{(g;g')} - (M-1)\hat{\rho}_{t;-r}^{(g;g')} \quad (r = 1, \dots, M). \quad (36)$$

Next we compute the average pseudo-value $\bar{J} = \sum J_r/M$, which is expected to have less bias. We use $s^2(\bar{J}) = s^2(J)/M$ where $s^2(\bar{J})$ is the estimated variance of this average pseudo-value:

$$s^2(\bar{J}) = \frac{\sum_{r=1}^M (J_r - \bar{J})^2}{(M-1)M}.$$

Finally, we compute a $1 - \alpha$ confidence interval for $\rho_t^{(g;g')}$:

$$[\bar{J} - t_{M-1;1-\alpha/2}s(\bar{J}), \bar{J} + t_{M-1;1-\alpha/2}s(\bar{J})]. \quad (37)$$

We obtain (mutually correlated) estimates $\hat{\rho}_t^{(1;2)}$ at the various input values t in each of our experiments, so when we use (35) or (37) we again apply Bonferroni's inequality— analogously to (i) and (ii). In all four experiments of our example we may accept the results.

(iv) Finally, we examine $\rho_{t,t'}^{(g)}(M)$, which denotes the *auto-correlation* between outputs g at locations t and t' estimated from M macro-replicates; see (33). Actually, we have m old input values and $m-1$ new values so altogether we have $2m-1$ outputs y_t ($t = 1, \dots, 2m-1$). Consequently, we have $2m-2$ observations $(y_t^{(g)}, y_{t+1}^{(g)})$ with the minimum distance (say) $h_1 = |x_t - x_{t+1}| = 1/(2m-1)$; as the distance h between the input values of the outputs increases, the number of observations decreases. This enables the following estimators of the covariances with distance h for output g :

$$\hat{c}_h^{(g)} = \frac{\sum_{t=1}^{2m-h-1} [y_{t;g} - \overline{y_{t;g}}][y_{t+h;g} - \overline{y_{t+h;g}}]}{2m-1} \quad (h = 0, 1, \dots, 2m-2) \quad (38)$$

where—because of the GP assumption—we use $\overline{y_{t;g}} = \overline{y_{t+h;g}} = \overline{y_g}$; we use the denominator $(2m-1)$ because MATLAB follows Box et al. (1994), who claim that this denominator

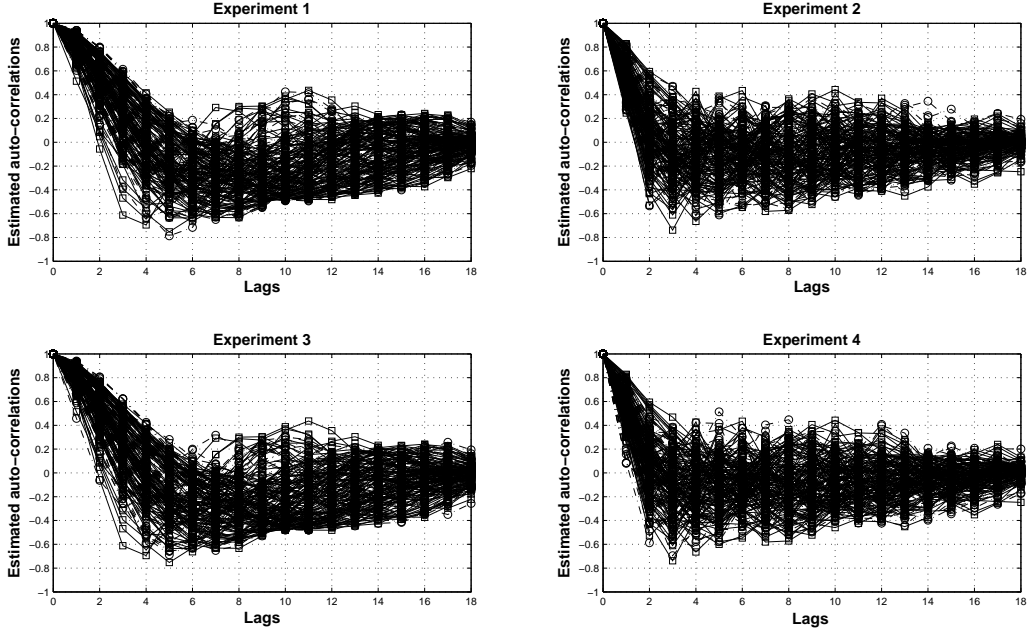


Figure 3: Estimated auto-correlations versus lags in 100 macro-replicates

minimizes the MSE (not the bias). So the auto-correlations are

$$\widehat{\rho}_h^{(g)} = \frac{\widehat{c}_h^{(g)}}{\widehat{c}_0^{(g)}}. \quad (39)$$

This equation gives Figure 3, which displays $\widehat{\rho}_{h;r}^{(g)}$ where r refers to macro-replicate r . This figure demonstrates that the low θ of experiments 1 and 3 does give a high estimated auto-correlation $\widehat{\rho}_h^{(g)}$ for small distances h ($h = 0, 1, 2$); for bigger distances, however, some auto-correlations become even negative. Note that high correlation implies that the sample gives relatively little information. Next we fit a curve to these $\widehat{\rho}_{h;r}^{(g)}$ per macro-replicate.

We apply nonlinear regression analysis using MATLAB, which follows Seber and Wild (2003). MATLAB fits a curve $\alpha_1 + \alpha_2 e^{-\theta_R h^2}$, which generalizes the Gaussian correlation function (defined in equation 2) through α_1 and α_2 (so we expect $\widehat{\alpha}_1 = 0$ and $\widehat{\alpha}_2 = 1$), resulting in the nonlinear regression estimate $\widehat{\theta}_{r;R}^{(g)}$ (which should be distinguished from the RMLE $\widehat{\theta}_r^{(g)}$). Our example gives the histogram with $M = 100$ values in Figure 4. The null-hypothesis $H_0 : \widehat{\theta}_{r;R}^{(g)} = \theta^{(g)}$ is rejected by both the t test and the sign test, but the difference $\widehat{\theta}_{r;R}^{(g)} - \theta^{(g)}$ is not really important. Similar conclusions hold for α_1 and α_2 . The $\widehat{\theta}_{r;R}^{(g)}$ does change much when the true value $\theta^{(g)}$ changes from 18 to 131.

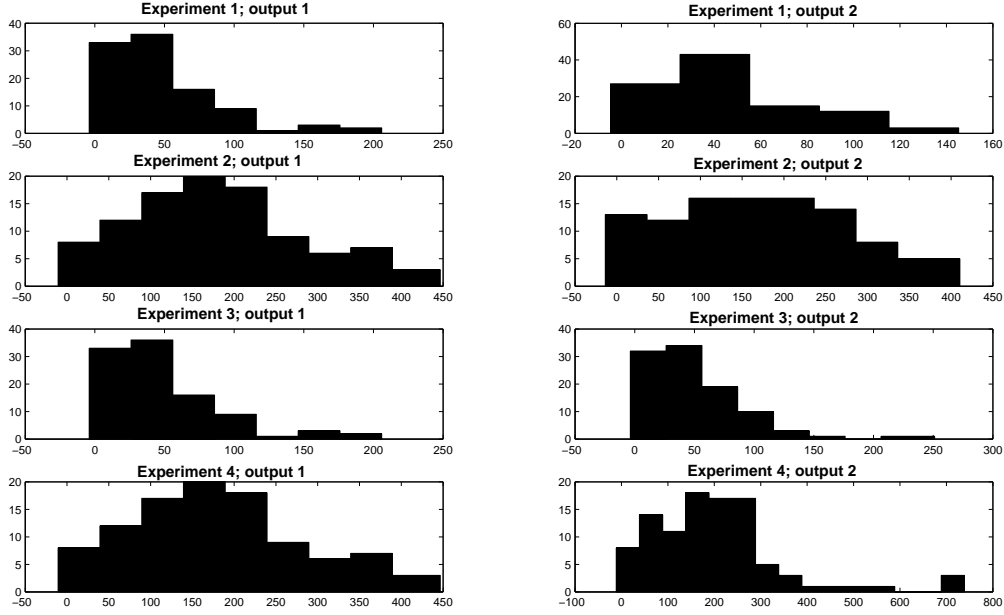


Figure 4: $\widehat{\theta}_{r;R}^{(g)}$ for all the four experiments; $g = 1$ and 2

We conclude that (i) the estimated auto-correlations (for different distances) are biased because they are ratio estimators and they use a denominator that minimizes MSE (not bias), (ii) these estimators are poor in case of high correlation. An alternative—for the tests detailed under (i) through (iv)—uses the RMLE that are standard in Kriging; see the next subsection.

4.2 Monte Carlo results

In practice, the simulation analysts have only “a single macro-replicate” to compute the RMLEs $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\Sigma}}$. These RMLEs are based on the likelihood function (24); this function may have many local maxima so the search for these RMLEs may get stuck on a local hill. To get initial estimates of $\widehat{\mathbf{A}}$ and $\widehat{\boldsymbol{\Theta}}$, Svenson (2011, pp. 314-315) uses a global optimizer; namely, the *Genetic Algorithm* (GA) in Forrester (2010); to get final estimates, he uses MATLAB’s “fmincon”. When we use DACE, we use the initial values of NSDep’s GA including his bounds.

Given these RMLEs (computed from the I/O data for the “old” points), we can predict the output for one of the new points (say) $x_{t'}$ with $t' = 1, \dots, m - 1$; i.e., in (3) we replace the unknown parameters by their RMLEs, which gives (say) $\widehat{y}_{t'}$. We compare this predicted

value with the “observed” value $y_{t'}$ conditional on the m old observed values y_i ; see (15) (because we generate this new observation using the true parameters, not their RMLEs). This gives the Squared Error (SE) at the new input value t' for output g in macro-replicate r :

$$SE_{t';r}^{(g)} = (\widehat{y_{t';r}^{(g)}} - y_{t';r}^{(g)})^2 \text{ with } (g = 1, \dots, n) (t' = 1, \dots, m - 1) (r = 1, \dots, M). \quad (40)$$

Our example has $n = 2$, $m = 10$, $M = 100$. Because we predict the output for $m - 1$ new input combinations, we use (40) to compute the estimated *Integrated MSE* (IMSE) for output g :

$$\widehat{IMSE}_r^{(g)} = \frac{\sum_{t'=1}^{m-1} SE_{t';r}^{(g)}}{m - 1}.$$

Next we use this $\widehat{IMSE}_r^{(g)}$ to compute the performance averaged over the M macro-replicates:

$$\overline{\widehat{IMSE}^{(g)}} = \frac{\sum_{r=1}^M \widehat{IMSE}_r^{(g)}}{M} \quad (41)$$

Table 2 gives (41) for DACE and NSDep, in our four experiments (these experiments were detailed in Table 1). DACE gives a smaller $\overline{\widehat{IMSE}^{(g)}}$ in six of the eight comparisons with NSDep.

Table 2: $\overline{\widehat{IMSE}^{(g)}}$ in DACE and NSDep, estimated from 100 macro-replicates

Experiment	Output 1		Output 2	
	DACE	NSDep	DACE	NSDep
1	0.000181	0.000238	0.004801	0.004784
2	0.239103	0.239563	5.762738	5.813574
3	0.000179	0.000198	0.004187	0.004012
4	0.239072	0.240658	5.712830	5.732354

To explain these $\overline{\widehat{IMSE}^{(g)}}$, we compute t -statistics from the $M = 100$ RMLEs for the various Kriging parameters $(\mu_g, \sigma_g^2, \theta^{(g)}, \sigma_{g;g'})$ where $\sigma_{g;g'} = 0$ in DACE. Table 3 shows whether a t -test rejects the null-hypothesis H_0 that states that the expected value of a RMLE equals the true value of the corresponding Kriging parameter; the superscript * denotes that the t -statistic is significant at the 5% significance level. This table suggests that for DACE we should reject H_0 for $\theta^{(g)}$; for NSDep we should reject H_0 for $\theta^{(g)}$, σ_g^2 , and $\sigma_{g;g'}$.

Svenson and Santner (2010) mention that RMLE in multivariate Kriging requires a search in higher dimensions than univariate Kriging does (which assumes zero cross-correlations),

Table 3: t -tests for RMLE in DACE and NSDep

DACE							
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	1.1510	1.1529	2.0268*	2.4019*	0.5672	-0.8968	-
2	1.4495	1.4557	-6.3886*	-4.9355*	-0.3692	-0.1543	-
3	1.1501	0.4909	1.9842	2.9331*	0.5652	-2.5660*	-
4	1.4502	0.7710	-6.3868*	-4.1674*	-0.3822	-1.5303	-
NSDep							
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	1.2284	1.2597	1.9355	-0.2059	2.4377*	2.2318*	2.1643*
2	1.4572	1.4027	-8.3775*	-8.6159*	3.1470*	2.8997*	2.9758*
3	0.8132	0.4204	0.3007	1.1068	1.2741	0.6756	1.5367
4	1.5185	0.8312	-9.1144*	-6.6839*	3.4481*	1.5533	1.9560

so the former search might actually result in poor estimates of the Kriging parameters. Svenson and Santner (2010) and Fricker et al. (2010) give numerical results for several examples suggesting that multivariate Kriging may not improve the accuracy dramatically relative to multiple application of univariate Kriging. To investigate these numerical results, we run DACE and NSDep with the *true* Kriging parameters. The “old” and “new” outputs vary over the M macro-replicates, because they are sampled from (25). Consequently, (41) gives $\widehat{IMSE}^{(g)}(\mu_g, \sigma_g^2, \theta^{(g)}, \sigma_{1;2})$; see Table 4 Comparing this table with Table 2 shows that $\widehat{IMSE}^{(g)}$ is smaller when using the true Kriging parameters instead of their RMLEs, as we expected.

Table 4: $\widehat{IMSE}^{(g)}$ in DACE and NSDep with true MLE parameters

Experiment	Output 1		Output 2	
	DACE	NSDep	DACE	NSDep
1	0.000123	0.000123	0.003318	0.003318
2	0.235071	0.235071	5.756298	5.756298
3	0.000123	0.000123	0.00299	0.00299
4	0.235071	0.235071	5.708968	5.708968

Moreover, Table 4 shows that DACE and NSDep give the same $\widehat{IMSE}^{(g)}$ when using the true Kriging parameters. To explain this phenomenon, we study the only difference between DACE and NSDep; namely, $\Sigma_{0;m;n} \Sigma_Y^{-1}$ in (15). DACE assumes zero cross-covariances.

We find that the corresponding elements in NSDep—using the true Kriging parameters—are virtually zero! For example, when predicting the output for the first element of $\mathbf{x}_0^T = (1/18, 3/18, \dots, 17/18)^T$ in macro-replicate 1 of experiment 1, NSDep gives values between 10^{-16} and 10^{-13} . We also compute these values for a cross-correlation 0.95 (instead of 0.8) and again find virtually zero values; see Table 7 in Appendix D.

Our conclusion is that DACE’s univariate code is simpler than NSDep’s multivariate code for the nonseparable dependence model, and that DACE gives the same IMSE estimated from 100 macro-replicates (other publications use only one macro-replicate; i.e., they apply RMLE to the given I/O data set).

Fricker et al. (2010) also study the *coverage* of the confidence interval of the Kriging predictor. A 90% confidence intervals for $\widehat{y_{t';r}^{(g)}}$ is

$$\widehat{y_{t';r}^{(g)}} - 1.64\sqrt{MSPE_{t';r}^{(g)}}, \widehat{y_{t';r}^{(g)}} + 1.64\sqrt{MSPE_{t';r}^{(g)}},$$

where 1.64 is the 0.95 quantile of the standard Gaussian distribution and $MSPE_{t';r}^{(g)}$ follows from (4) and (17). Obviously, for given t' and g values, macro-replicate r gives an interval that does or does not cover the true value $y_{t';r}^{(g)}$; from the M macro-replicates we compute the estimated coverage. The coverage of this 90% confidence interval turns out to be too low, for any t and g (box plots are available from the authors). This low coverage may be explained through the classic variance of the Kriging predictor, which ignores the fact that the Kriging parameters are not known but are estimated (through RMLE); see again den Hertog et al. (2006).

Finally, Fricker et al. (2010, p. 15) suggest that the relative performance of multivariate Kriging may improve when “the” output is a function of the individual (cross-correlated) outputs. We therefore add experiments with the sum and the product, $y^{(3)} = y^{(1)} + y^{(2)}$ and $y^{(4)} = y^{(1)}y^{(2)}$. Appendix E shows that DACE gives smaller $\widehat{IMSE}^{(g)}$ for all $n = 4$ outputs and all four experiments except for $y^{(4)}$ in experiment 1. DACE’s coverage is better than NSDep’s, but still below the nominal value. Note that we do know the true values of the Kriging parameters for $y^{(1)}$ and $y^{(2)}$, but not for $y^{(3)}$ and $y^{(4)}$. Furthermore, there are more Kriging parameters to be estimated; namely, $\sigma_{g,g'}.(g, g' = 1, \dots, n)$ and $\theta^{(g)}$.

5. Conclusions and future research

In this paper we formalize multivariate Kriging or GP metamodels applied to random simulation models with multiple outputs. A major problem of multivariate GP is to ensure that its covariance matrix is positive-definite; to solve this problem, we use nonseparable dependence model. We compare this multivariate Kriging with univariate Kriging applied per type of simulation output, which ignores the (cross)correlations between simulation outputs for a given input combination. To evaluate the performance of these two Kriging types, we construct a Monte Carlo laboratory that guarantees that all the assumptions of multivariate Kriging are satisfied. To verify that this laboratory works as intended, we derive statistics that enable us to test whether the Kriging parameters have the correct (“true”) values. We use this (verified) laboratory to quantify the performance of multivariate and univariate Kriging in the “simplest” case; namely; a single input and two outputs that have a GP with Gaussian auto-correlation functions and specific cross-correlation values. To measure this performance, we use the IMSE; we also consider the coverage of the classic confidence interval for the Kriging predictor. For this case we estimate that univariate Kriging has a smaller IMSE and higher coverage; one explanation is that multivariate Kriging requires RMLEs for several more Kriging parameters; namely, the cross-correlations. To examine this explanation, we run Monte Carlo experiments replacing these RMLEs by the true values of the Kriging parameters. We find that multivariate and univariate Kriging then give virtually the same predictors, and hence the same IMSE. We briefly examine the coverage, and conclude that both multivariate and univariate Kriging give coverages lower than the nominal (90%) value; multivariate Kriging does not improve this coverage.

Future research may address the following problems.

- We showed numerically that multivariate Kriging does not outperform univariate Kriging. In future research we might be able to prove analytically that both types of Kriging become the same if the multiple outputs are generated by the same input matrix, analogously to Rao (1967)’s proof for linear regression.
- Multivariate Kriging may serve as a metamodel not only for simulation models with multiple outputs but also for situations in which “quick and dirty” simulation models are run for many input combinations, whereas slow but accurate simulations are run

for fewer combinations; see Santner et al. (2003) and Forrester et al. (2008) and also “partially heterotropic” situations in Wackernagle (2003, p. 158).

- Universal Kriging (instead of Ordinary Kriging) may improve the validity of the GP metamodel; see Fricker et al. (2010).
- The Gaussian correlation function may be replaced by some other correlation function; see Lophaven et al. (2002b, p. 6).
- RMLE requires nonlinear optimization, and is a challenging mathematical problem. For example, we may follow Xia et al. (2008, p. 936) and replace the parameters $\hat{\sigma}_z^2$ and $\hat{\theta}_j$ by the monotonic transformations $\exp(\hat{\sigma}_z^2)$ and $\exp(\hat{\theta}_j)$ because these parameters must be positive—which makes it a constrained optimization problem; however, after these transformations the problem becomes an unconstrained optimization problem, which is easier to solve. For the resulting unconstrained optimization, Xia et al. (2008) use the MATLAB function “fminunc”. Couckuyt et al. (2010) and Gano et al. (2006) also discuss heuristics for RMLE computation.
- Because multivariate Kriging does not outperform univariate Kriging in deterministic simulation, it does not seem interesting to extend multivariate Kriging from deterministic to random simulation.

Acknowledgments

We thank Joshua Svenson and Tom Santner (Ohio State) for sharing their code with us.

References

- Antognini, A. B., M. Zagoraiou. 2010. Exact optimal designs for computer experiments via kriging metamodeling. *J. of Statist. Planning and Inference* **140** 2607–2617. doi: 10.1016/j.jspi.2010.03.027.
- Box, G.E.P., G.M. Jenkins, G.C. Reinsel. 1994. *Time series analysis: forecasting and control*. Forecasting and Control Series, Prentice Hall.
- Boyle, P., M. R. Frean. 2004. Dependent gaussian processes. *Neural Information Processing Systems*.

- Chan, W.K., L. W. Schruben, B. L. Nelson, S. H. Jacobson. 2009. Simulation fusion. M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, R. G. Ingalls, eds., *Winter Simulation Conf.*. 686–695. doi:10.1109/WSC.2009.5429682.
- Chen, X., B. Ankenman, B.L. Nelson. 2010. The effects of common random numbers on stochastic kriging metamodels. Tech. rep., Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL USA.
- Couckuyt, I., D. Gorissen, T. Dhaene, A. Forrester. 2010. Blind kriging: implementation and performance analysis. Tech. rep., Ghent University, Department of Information Technology (INTEC), Ghent, Belgium.
- Cressie, N. 1991. *Statistics for spatial data*. Wiley, New York.
- den Hertog, D., J.P.C Kleijnen, A.Y.D Siem. 2006. The correct kriging variance estimated by bootstrapping. *J. of The Oper. Res. Soc.* **57** 400–409. doi:10.1057/palgrave.jors.2601997.
- Forrester, A. I. J. 2010. Black-box calibration for complex-system simulation. *Philos. Trans. of The Roy. Soc. A: Math., Phys. and Engrg. Sci.* **368** 3567–3579. doi:10.1098/rsta.2010.0051.
- Forrester, Alexander I. J., Andras Sobester, Andy J. Keane. 2008. *Engineering Design via Surrogate Modelling - A Practical Guide*. Wiley, Chichester, U.K. doi:10.1002/9780470770801.
- Frazier, P.I. 2010. *Wiley Encyclopedia of Operations Research and Management Science*, chap. Learning with Dynamic Programming. Wiley, New York, 1–13. In press.
- Fricke, T.E., J. E. Oakley, N. M. Urban. 2010. Multivariate emulators with nonseparable covariance structures. Tech. rep., Managing Uncertainty in Complex Models (MUCM).
- Gano, S. E., J. E. Renaud, J. D. Martin, T. W. Simpson. 2006. Update strategies for kriging models used in variable fidelity optimization. *Structural and Multidisciplinary Optim.* **32** 287–298. doi:10.1007/s00158-006-0025-y.
- Gneiting, T., W. Kleiber, M. Schlather. 2010. Matern cross-covariance functions for multivariate random fields. *J. of the Amer. Statist. Association* **105** 1167–1177. doi:10.1198/jasa.2010.tm09420.

- Higdon, D., J. Gattiker, B. Williams, M. Rightley. 2008. Computer model calibration using high-dimensional output. *J. of the Amer. Statist. Association* **103** 570–583. doi:10.1198/016214507000000888.
- Iacoa, S. De, D.E. Myersb, D. Posa. 2011. On strict positive definiteness of product and product-sum covariance models. *J. of Statist. Planning and Inference* **141** 1132–1140.
- Jones, D. R., M. Schonlau, W. J. Welch. 1998. Efficient global optimization of expensive black-box functions. *J. of Global Optim.* **13** 455–492.
- Karplus, W. J. 1983. The spectrum of mathematical models. *Perspectives in Comput.* **3** 4–13.
- Kleijnen, J. P. C. 1993. Simulation and optimization in production planning a case study. *Decision Support Systems* **9** 269–280.
- Kleijnen, J. P. C. 2008. *Design and analysis of simulation experiments*. Springer-Verlag. (Chinese translation published by Publishing House of Electronics Industry, Beijing, 2010).
- Kleijnen, J. P. C., R. G. Sargent. 2000. A methodology for fitting and validating metamodels in simulation. *Eur. J. of Oper. Res.* **120** 14–29.
- Kleijnen, J. P. C., M. T. Smits. 2003. Performance metrics in supply chain management. *J. of The Oper. Res. Soc.* **54** 507–514. doi:10.1057/palgrave.jors.2601539.
- Kleijnen, J. P. C., W. van Beers, I. van Nieuwenhuysse. 2010. Constrained optimization in simulation: a novel approach. *Eur. J. of Oper. Res.* **202** 164–174.
- Kleijnen, J. P. C., W. van Beers, I. van Nieuwenhuysse. 2011. Expected improvement in efficient global optimization through bootstrapped kriging. *J. of Global Optim.* doi:10.1007/s10898-011-9741-y. Accepted.
- Kleijnen, J. P. C., G. van Ham, J. Rotmans. 1992. Techniques for sensitivity analysis of simulation models: A case study of the CO_2 greenhouse effect. *SIMULATION* **58** 410–417. doi:10.1177/003754979205800610.
- Li, G., S. Azarm, A. Farhang-Mehr, A. R. Diaz. 2006. Approximation of multiresponse deterministic engineering simulations: a dependent metamodeling approach. *Structural and Multidisciplinary Optim.* **31** 260–269. doi:10.1007/s00158-005-0574-5.

- Li, R., A. Sudjianto. 2005. Analysis of computer experiments using penalized likelihood in gaussian kriging models. *Technometrics* **47** 111–120. doi:10.1198/004017004000000671.
- Loeppky, J. L., J. Sacks, W. J. Welch. 2009. Choosing the sample size of a computer experiment: A practical guide. *Technometrics* **51** 366–376. doi:10.1198/TECH.2009.08040.
- Lophaven, S.N., H.B. Nielsen, J. Sondergaard. 2002a. *Aspects of the MATLAB toolbox DACE*. IMM Technical University of Denmark, Lyngby, Denmark.
- Lophaven, S.N., H.B. Nielsen, J. Sondergaard. 2002b. *DACE: a MATLAB Kriging toolbox, version 2.0*. IMM Technical University of Denmark, Lyngby, Denmark.
- Mahevas, S., D. Pelletier. 2004. Isis-fish, a generic and spatially explicit simulation tool for evaluating the impact of management measures on fisheries dynamics. *Ecological Model.* **171** 65 – 84. doi:10.1016/j.ecolmodel.2003.04.001.
- Marrel, A., B. Iooss, S. da Veiga, M. Ribatet. 2010. Global sensitivity analysis of stochastic computer models with joint metamodels. *Statist. and Comput.* **22** 833–847.
- Press, W.H., S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. 1992. *Numerical Recipes in C. The Art of Scientific Computing*, Cambridge University Press.
- Rao, C.R. 1967. Least squares theory using an estimated dispersion matrix and its application to measurement of signals. L.M. Le Cam, J. Neyman, eds., *Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. I. University of California Press, Berkeley, CA., 355–372.
- Santner, T. J., B. J. Williams, W. I. Notz. 2003. *The Design and Analysis of Computer Experiments*. Springer-Verlag, New York.
- Seber, G.A.F., C.J. Wild. 2003. *Nonlinear Regression*. Wiley Series in Probability and Statistics, Wiley-Interscience.
- Simpson, T.W., J.D. Poplinski, P. N. Koch, J.K. Allen. 2001. Metamodels for computer-based engineering design: Survey and recommendations. *Engrg. with Comput.* **17** 129–150.
- Svenson, J.D. 2011. Computer experiments: multiobjective optimization and sensitivity analysis. Ph.D. thesis, The Ohio State University, The Ohio State University.

- Svenson, J.D., T.J. Santner. 2010. Multiobjective optimization of expensive black-box functions via expected maximin improvement. Tech. Rep. 43210, Ohio University, Columbus, Ohio.
- Wackernagle, H. 2003. *Multivariate geostatistics: an introduction with applications*. Springer-Verlag, New York.
- Williams, B.J., T.J. Santner, W.I. Notz, J.S. Lehman. 2010. *Statistical Modelling and Regression Structures - Festschrift in the Honour of Ludwig Fahrmeir*, chap. Sequential design of computer experiments for constrained optimization. Springer-Verlag, 449–472.
- Xia, H., Y. Ding, J. Wang. 2008. Gaussian process method for form error assessment using coordinate measurements. *IIE Trans.* **40** 931–946. doi:10.1080/07408170801971502.
- Yaglom, A. 1987. *Correlation Theory of Stationary and Related Random Functions I: Basic Results*. Springer-Verlag, New York.

A. List of major symbols

A list of major symbols is given in Table 5, in alphabetical order with Latin symbols before Greek symbols; bold letters denote matrixes and vectors.

B. I/O data for multivariate Kriging

Table 6 shows the I/O data for multivariate Kriging.

C. Simplest example: two outputs and one input

The simplest example of a simulation model with multiple types of outputs is a model with *two* types only; say, y_1 and y_2 . Furthermore, the simplest model has a *single* input (say) x . Figure 5 illustrates this simplest model, which is simulated for $m = 3$ input values. Two input values are relatively close together; namely, x_1 and x_2 (x_3 is relatively far away from x_1 and x_2). If we consider the bivariate output (y_1, y_2) at a given input value such as x_1 , then we see that these two outputs $y_1(x_1)$ and $y_2(x_1)$ are “cross-correlated”; Figure 5 shows this correlation through the vertical dotted curves. Moreover, the Kriging metamodel implies

Table 5: List of major symbols

Symbol	Meaning
$c_{0,i}$	covariance between outputs of old input combination i and new combination 0
i	index with range $1, \dots, m$
g	index with range $1, \dots, n$
k	number of simulation inputs
m	number of “old” simulated input combinations
n	number of output types per input combination
\mathbf{R}	correlation matrix
t	index with range $1, \dots, 2m - 1$
t'	index with range $1, \dots, m - 1$
\mathbf{x}	input combination
\mathbf{x}_0	new input combination
y	univariate output of a simulated input combination
\mathbf{Y}	multivariate output of a simulated input combination
$y_{i;g}$	output of type g for input combination i
\hat{y}_0	univariate Kriging predictor of output of new input combination x_0
z	stationary Gaussian process with zero mean
$\theta_j^{(g)}$	importance of input j for the auto-correlation in outputs g
μ	mean univariate output
μ_g	mean of output type g
$\rho_{i,i'}^{(g;g')}$	correlation between outputs g and g' at input combinations i and i'
$\sigma_{i,i'}^{(g;g')}$	covariance between outputs g and g' at input combinations i and i'
σ_z^2	variance of univariate z
Σ	covariance matrix of univariate output
Σ_0	covariance of \mathbf{Y}
$\Sigma_{0;m;n}$	$n \times nm$ covariance between $\mathbf{Y}(\mathbf{x}_0)$ and \mathbf{Y}
$\Sigma_{\mathbf{Y}}$	covariance matrix of multivariate output \mathbf{Y}

Table 6: I/O data for multivariate Kriging

Combination	Input	Output
1	$x_{1;1}, \dots, x_{1;k}$	$y_{1;1}, \dots, y_{1;n}$
2	$x_{2;1}, \dots, x_{2;k}$	$y_{2;1}, \dots, y_{2;n}$
\vdots	\vdots	\vdots
i	$x_{i;1}, \dots, x_{i;k}$	$y_{i;1}, \dots, y_{i;n}$
\vdots	\vdots	\vdots
m	$x_{m;1}, \dots, x_{m;k}$	$y_{m;1}, \dots, y_{m;n}$

that a given type of output such as y_1 is correlated with itself when observed at different input combinations; e.g., $y_1(x_1)$ and $y_1(x_2)$ are correlated. This correlation we call “auto-correlation”. The figure shows this auto-correlation through the (tilted horizontal) dotted lines. The other type of output y_2 is also auto-correlated, but we do not show this correlation in the figure, to keep the figure simple. Obviously, outputs such as $y_1(x_3)$ and $y_2(x_1)$ are also correlated: in Figure 5 we can follow the line from $y_1(x_3)$ and $y_1(x_1)$, and then the vertical curve to $y_2(x_1)$.

Mathematically, for our simplest example the Kriging metamodel with the Gaussian correlation function (2) for output of type 1 becomes

$$\text{corr}[y_1(x_i), y_1(x_j)] = \exp[-\theta^{(1)}(x_i - x_j)^2] \text{ with } \theta^{(1)} \geq 0 \quad (42)$$

where we suppress the subscript $j = 1$ of x and $\theta^{(1)}$ in (2) because there is only one input. In the example the input x has values such that the outputs of type 1 $y_1(x_1)$ and $y_1(x_2)$ have higher positive (auto)correlation than $y_1(x_1)$ and $y_1(x_3)$ have. Notice that if the input has the same value $x_i = x_j = x$ in (42), then

$$\text{corr}[y_1(x), y_1(x)] = \exp[-\theta^{(1)}(x - x)^2] = \exp(-\theta^{(1)} \times 0) = \exp(-0) = 1/1 = 1, \quad (43)$$

whatever the value of $\theta^{(1)}$ is. So, these outputs have the highest positive (auto)correlation— which makes perfect sense in deterministic simulation. For output of type 2 we replace $\theta^{(1)}$ by $\theta^{(2)}$ in (42).

Note: Stationarity of the process implies that only the distance in the input space matters, not the direction; e.g., $y_{1;1}$ and $y_{2;2}$ are as strongly correlated as $y_{2;1}$ and $y_{1;2}$ are

D. $\Sigma_{0;m;n} \Sigma_Y^{-1}$ in DACE and NSDep

Table 7 gives $\Sigma_{0;m;n} \Sigma_Y^{-1}$ when predicting the output for $x_0 = 1/18$ in macro-replicate 1 of experiment 1, for cross-correlations 0.80 and 0.95; to improve the layout we present the transpose of this matrix.

E. Kriging for functions of outputs

Table 8 shows \widehat{IMSE} for $y^{(1)}$, $y^{(2)}$, $y^{(3)} = y^{(1)} + y^{(2)}$, and $y^{(4)} = y^{(1)}y^{(2)}$. Table 9 gives the t statistics to test whether the RMLEs of the Kriging parameters significantly differ from the true values for $y^{(1)}$ and $y^{(2)}$; for the other two outputs ($y^{(3)}$ and $y^{(4)}$) we do not know the true parameters, so we cannot apply these tests.

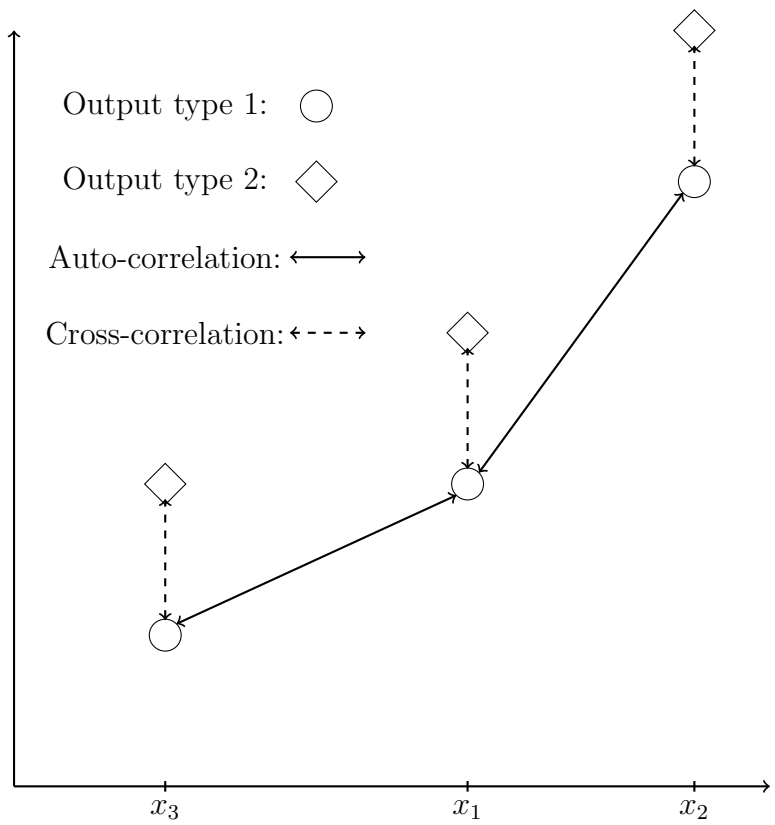


Figure 5: Simplest example: bivariate output and single input

Table 7: $\Sigma_{0;m;n}\Sigma_Y^{-1}$ in NSDep and DACE

		NSDep		DACE	
		$\rho = 0.8$	$\rho = 0.95$	$\rho = 0.8$	
0.348906	-2.6E-14	0.348906	-2.7E-13	0.348906	0
-3.3E-16	0.348906	-9.4E-15	0.348906	0	0.348906
0.954562	1.6E-13	0.954562	0	0.954562	0
-4.9E-15	0.954562	-1.6E-14	0.954562	0	0.954562
-0.51635	-1.3E-13	-0.51635	4.55E-13	-0.51635	0
7.11E-15	-0.51635	7.13E-14	-0.51635	0	-0.51635
0.389122	-1.1E-13	0.389122	-2.3E-13	0.389122	0
-1E-14	0.389122	-1E-13	0.389122	0	0.389122
-0.29666	2.22E-13	-0.29666	-1.6E-12	-0.29666	0
1.06E-14	-0.29666	2.85E-14	-0.29666	0	-0.29666
0.216743	1.99E-13	0.216743	-6.8E-13	0.216743	0
-1.2E-14	0.216743	1.24E-14	0.216743	0	0.216743
-0.14675	1.11E-13	-0.14675	-4.8E-13	-0.14675	0
7.77E-15	-0.14675	-1.1E-14	-0.14675	0	-0.14675
0.087582	6E-14	0.087582	-2.1E-13	0.087582	0
-1.9E-15	0.087582	1.07E-14	0.087582	0	0.087582
-0.04163	-6.9E-15	-0.04163	1.14E-13	-0.04163	0
-1.3E-15	-0.04163	3.52E-15	-0.04163	0	-0.04163
0.012026	-1.1E-16	0.012026	-1.4E-14	0.012026	0
-1.2E-16	0.012026	2.21E-15	0.012026	0	0.012026

Table 8: \widehat{IMSE} in DACE and NSDep

Experiment	Output 1		Output 2	
	DACE	NSDep	DACE	NSDep
1	0.000181	0.000915	0.0048008	0.011003
2	0.239103	0.245076	5.7627378	5.862312
3	0.000179	0.000748	0.0041866	0.011534
4	0.239072	0.246433	5.7128304	5.894214
Experiment	Output 3		Output 4	
	DACE	NSDep	DACE	NSDep
1	0.006536	0.016618	0.290038	0.241913
2	7.880984	8.020599	26.83617	28.45734
3	0.004553	0.01359	0.14392	0.152387
4	6.403224	6.576276	16.85455	18.1546

Table 9: t -tests for RMLE in DACE and NSDep; four outputs

DACE							
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	1.1510	1.1529	2.0268*	2.4019*	0.5672	-0.8968	-
2	1.4495	1.4557	-6.3886*	-4.9355*	-0.3692	-0.1543	-
3	1.1501	0.4909	1.9842	2.9331*	0.5652	-2.5660*	-
4	1.4502	0.7710	-6.3868*	-4.1674*	-0.3822	-1.5303	-
NSDep							
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	0.8560	1.0006	9.0147*	3.7084*	1.3071	3.5246*	1.5648
2	1.4535	1.1069	-16.8047*	-15.6625*	3.3923*	6.2856*	2.3446*
3	0.8557	0.7707	5.8053*	4.0578*	3.4747*	3.4712*	2.2469*
4	1.7358	0.4621	-19.1027*	-17.0154*	5.9945*	5.1180*	0.8354