Supplementary materials for this article are available online. Please go to http://www.tandfonline.com/r/TECH

Computer Experiments With Both Qualitative and Quantitative Variables

Hengzhen HUANG, Min-Qian LIU, and Jian-Feng YANG

LPMC and Institute of Statistics, Nankai University
Tianjin 300071, China
(*jfyang@nankai.edu.cn*)

Dennis K. J. LIN

Department of Statistics
The Pennsylvania State University
University Park, PA 16802

Computer experiments have received a great deal of attention in many fields of science and technology. Most literature assumes that all the input variables are quantitative. However, researchers often encounter computer experiments involving both qualitative and quantitative variables (BQQV). In this article, a new interface on design and analysis for computer experiments with BQQV is proposed. The new designs are one kind of sliced Latin hypercube designs with points clustered in the design region and possess good uniformity for each slice. For computer experiments with BQQV, such designs help to measure the similarities among responses of different level-combinations in the qualitative variables. An adaptive analysis strategy intended for the proposed designs is developed. The proposed strategy allows us to automatically extract information from useful auxiliary responses to increase the precision of prediction for the target response. The interface between the proposed design and the analysis strategy is demonstrated to be effective via simulation and a real-life example from the food engineering literature. Supplementary materials for this article are available online.

KEY WORDS: Cross-validation; Gaussian process model; Kriging; Latin hypercube design; Similarity.

1. INTRODUCTION

Computer experiments are becoming increasingly prevalent surrogates for physical experiments (Santner, Williams, and Notz 2003; Fang, Li, and Sudjianto 2006). Most relevant research has paid attention to situations where all the input variables are quantitative. However, researchers often encounter computer experiments with both qualitative and quantitative variables (BQQV). See, for examples, Rawlinson et al. (2006), Long and Bartel (2006), Qian et al. (2006), Qian and Wu (2008), Qian, Tang, and Wu (2009), among others.

The modeling methods for computer experiments with BQQV are more complicated than those involving only quantitative variables. Most efforts have attempted to connect the information of all the responses corresponding to different level-combinations of the qualitative variables. McMillian et al. (1999), Joseph and Delaney (2007) and Zhou, Qian, and Zhou (2011) proposed prediction methods based on Gaussian process (GP) models. Han et al. (2009), from a different aspect, proposed a hierarchical Bayesian model. Herein, the response to be predicted is called the target response (TR) and the remaining ones are called the auxiliary responses (ARs). The methods mentioned above use the information of all the ARs to predict a TR. However, if an AR is not similar to the TR, the information of such an AR may reduce the prediction accuracy of the TR. Han et al. (2010) confirmed this point and suggested the ANOVA kriging model to select the similar responses. One limitation of the ANOVA kriging is that only one response can be set as the TR because the ANOVA kriging calls for a different design if the TR is changed; while practical situations often require setting each response in turn as the TR with the design unchanged (see Qian, Wu, and Wu 2008; Zhou, Qian, and Zhou 2011). Note that the term "different responses" here means "the responses corresponding to different level-combinations of the qualitative variables." That is, we consider a single response whose output vector can be partitioned into slices corresponding to different level-combinations of the qualitative variables. This is different from the concept of "multiple responses."

A new interface on design and analysis for computer experiments with BQQV is studied. The new design, called an optimal clustered-sliced Latin hypercube design (OCSLHD), is proposed to obtain the similarity measures among different responses. Based on the similarity measures produced by the OCSLHD, an analysis strategy is developed to select the ARs that are similar to the TR. Then, these ARs are included into the models for predicting the TR. That is, the interface between the proposed design and analysis helps to remove useless information for the TR. This is ignored by the existing design and analysis framework for computer experiments with BQQV (except perhaps the ANOVA kriging method). By discarding the useless ARs, the prediction accuracy of the TR is improved. The new interface on design and analysis is applied to the models of McMillian et al. (1999), Joseph and Delaney (2007) and Zhou, Qian, and Zhou (2011). Unlike the ANOVA kriging, the proposed framework here is able to set each response in turn as the TR under the same design.

This article is organized as follows. Section 2 introduces the underlying models, followed by a discussion on when an AR is useful for predicting the TR. Section 3 presents the design construction method. Section 4 develops an analysis strategy

TECHNOMETRICS, XXX 2016, VOL. XX, NO. XX
DOI: 10.1080/00401706.2015.1094416
Color versions of one or more of the figures in the article can be found online at *www.tandfonline.com/r/tech*.

intended for the proposed designs. Two simulated examples and a real-life example are studied to demonstrate the effectiveness of the interface between the proposed design and analysis strategy in Sections 5 and 6, respectively. Concluding remarks are given in Section 7. Some additional contents mentioned in the main text are placed in the supplementary material, which is available online.

2. METAMODELS AND THE CONDITION FOR AN AR BEING USEFUL

In this section, the metamodeling framework for computer experiments with BQQV will be introduced. We then discuss the question of when an AR will be useful for predicting the TR.

2.1 Gaussian Process Models With BQQV

A metamodel of a computer experiment seeks to represent the relation between the output and input variables well. The GP model plays a critical role for metamodeling due to its convenience, flexibility, and broad generality. The GP model used in most of the literature involves only quantitative variables. This section briefly describes how to use the GP model to build a metamodel with BQQV. For more details, refer to Qian, Wu, and Wu (2008).

Suppose that for $u = 1, \ldots, N$, $\mathbf{x}_u = (x_{u1}, \ldots, x_{ut})^T$ and $\mathbf{z}_u = (z_{u1}, \ldots, z_{ul})^T$ are the uth inputs of quantitative and qualitative variables, respectively. For $j = 1, \ldots, l$, assume the jth qualitative variable has q_j levels, denoted by $1, \ldots, q_j$. Let $s = \prod_{j=1}^{l} q_j$, and denote the s level-combinations of the qualitative variables by $c_1, \ldots, c_s$. Let $\mathbf{w}_u = (\mathbf{x}_u^T, \mathbf{z}_u^T)^T$ be the uth input vector and $y(\mathbf{w}_u)$ be the output value at $\mathbf{w}_u$. Then, $y(\mathbf{w}_u)$ can be expressed as $y(\mathbf{w}_u) = \mathbf{f}(\mathbf{w}_u)^T\boldsymbol{\beta} + \epsilon(\mathbf{w}_u)$, $u = 1, \ldots, N$, where $\mathbf{f}(\mathbf{w}) = (f_1(\mathbf{w}), \ldots, f_p(\mathbf{w}))^T$ is a set of known functions, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is the vector of unknown coefficients. $\mathbf{f}(\mathbf{w}_u)^T\boldsymbol{\beta}$ is called the regression part, and $\epsilon(\mathbf{w}_u)$ is the residual part which is assumed to be a zero-mean stationary GP with covariance function

$$\operatorname{cov}(\epsilon(\mathbf{w}_u), \epsilon(\mathbf{w}_v)) = \sigma^2\varphi_1(\mathbf{x}_u, \mathbf{x}_v)\varphi_2(\mathbf{z}_u, \mathbf{z}_v), \quad \text{for } u, v = 1, \ldots, N, \tag{1}$$

where σ^2 is the variance, $\varphi_1(\mathbf{x}_u, \mathbf{x}_v)$ and $\varphi_2(\mathbf{z}_u, \mathbf{z}_v)$ are the correlation functions for the quantitative variables and qualitative variables, respectively. The most commonly used form for $\varphi_1(\mathbf{x}_u, \mathbf{x}_v)$ is the Gaussian correlation function

$$\varphi_1(\mathbf{x}_u, \mathbf{x}_v) = \prod_{k=1}^{t} \exp\{-\theta_k|x_{uk} - x_{vk}|^2\}, \tag{2}$$

where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_t)^T$ is the vector of roughness parameters with each element being positive. Throughout this article, (2) will be used for $\varphi_1(\mathbf{x}_u, \mathbf{x}_v)$. There are two popular ways to model $\varphi_2(\mathbf{z}_u, \mathbf{z}_v)$. One of them is of the form

$$\varphi_2(\mathbf{z}_u, \mathbf{z}_v) = \tau_{c_u^*, c_v^*}, \tag{3}$$

where c_u^* represents the level-combination of the qualitative variables in $\mathbf{z}_u$ ($c_u^* \in \{c_1, \ldots, c_s\}$, and $\tau_{c_u^*, c_v^*}$ is the cross-correlation between responses corresponding to level-combinations c_u^* and c_v^*. As proved by Qian, Wu, and Wu (2008), (3) is a valid correlation function provided that the $s \times s$ matrix $\mathbf{P} = (\tau_{c_i, c_j})$ is a positive-definite matrix with unit diagonal elements (PDUDE). Several choices of the τ_{c_i, c_j} in the literature satisfy this condition. Joseph and Delaney (2007) suggested $\tau_{c_i, c_j} = c$ $(0 < c < 1)$ for $i \neq j$, called the exchangeable correlation (EC) function. McMillian et al. (1999) suggested $\tau_{c_i, c_j} = \exp\{-(\phi_i + \phi_j)I(i \neq j)\}$ $(\phi_i, \phi_j > 0)$, called the multiplicative correlation (MC) function. Zhou, Qian, and Zhou (2011) modeled the matrix $\mathbf{P}$ by using the hypersphere decomposition based unrestrictive correlation (UC) function. These three models are all called the integral kriging models. Another way to model $\varphi_2(\mathbf{z}_u, \mathbf{z}_v)$ is to use a product form of (3) (Santner, Williams, and Notz 2003), that is,

$$\varphi_2(\mathbf{z}_u, \mathbf{z}_v) = \left[\prod_{j=1}^{l} \tau^{(j)}_{z_{uj}, z_{vj}}\right], \tag{4}$$

where $\mathbf{P}_j = (\tau^{(j)}_{r,m})$ $(r, m = 1, \ldots, q_j)$ is a $q_j \times q_j$ PDUDE. Applying (4) instead of (3) may significantly reduce the number of parameters. A detailed discussion on the parameters involved in (3) and (4) is given in Supplementary Section S1. In sum, (4) has no more parameters than (3), except for the EC model, when (4) has more parameters. Note that no matter whether the correlation function is of form (3) or (4), it is only the level-combinations that matter for a computer experiment with BQQV, that is, each level-combination of the qualitative variables determines its own response surface.

An unknown constant μ is used as the regression part throughout this work. Such a GP model is known as *ordinary kriging* in the literature. Let $\mathbf{R}$ be the $N \times N$ matrix with the (u, v)th element being $\varphi_1(\mathbf{x}_u, \mathbf{x}_v)\varphi_2(\mathbf{z}_u, \mathbf{z}_v)$. The best linear unbiased predictor (BLUP) of an ordinary kriging model at an untried site $\mathbf{w}^*$ is

$$\hat{y}(\mathbf{w}^*) = \mu + \mathbf{r}^T\mathbf{R}^{-1}(\mathbf{y} - \mathbf{1}_N\mu), \tag{5}$$

where $\mu = (\mathbf{1}_N^T\mathbf{R}^{-1}\mathbf{1}_N)^{-1}\mathbf{1}_N^T\mathbf{R}^{-1}\mathbf{y}$, $\mathbf{y} = (y_1, \ldots, y_N)^T$ and $\mathbf{r} = (r(\mathbf{w}^*, \mathbf{w}_1), \ldots, r(\mathbf{w}^*, \mathbf{w}_N))^T$. In (5), $\mathbf{R}$, $\mathbf{r}$ and μ depend on the correlation function $r(\cdot)$, which in turn depends on the roughness parameter $\boldsymbol{\theta}$ in (2) and the cross-correlation parameters in (3) or (4). The most popular approach to estimate them is the maximum likelihood method and some powerful optimization algorithms can be found in Fang, Li, and Sudjianto (2006).

2.2 Similarity

The GP model framework indicates that an AR is useful for predicting the TR if both of them satisfy the model assumptions (1)–(5). In addition, a useful AR should possess a "large" (in absolute value) cross-correlation with the TR. Theoretically speaking, an AR that has a cross-correlation close to 0 with the TR would not affect the prediction of the TR. However, since such an AR provides little information for the TR, we recommend its removal, eliminating unnecessary parameter estimation. The following four conditions can be used to determine whether an AR is useful for the TR, where the first three are

related to the model assumptions and the last one is related to the cross-correlation.

1. The overall mean of the AR is close to that of the TR (a common regression part μ).
2. The amplitude of the variation in the AR is close to that in the TR (a common process variance σ^2).
3. The frequency of the oscillation for the AR is close to that for the TR (a common roughness parameter $\boldsymbol{\theta}$).
4. The phase of the oscillation for the AR is nearly the same (or opposite) as that for the TR (a large cross-correlation).

If the above four conditions are satisfied, the AR is useful; otherwise, it should not be used for predicting the TR. Note that Conditions 1 and 2 are not essential since the differences of the overall means and the amplitudes of the variation can be addressed by normalizing the response data (subtract the mean and divide by the standard deviation) for each response surface when *space-filling* designs are adopted (Santner, Williams, and Notz 2003). Data normalization thus is recommended as part of any computer experiment with BQQV. Conditions 3 and 4 form the basis for deciding whether an AR is useful. Specifically, when Condition 3 is violated but a common roughness parameter is assumed, predictions for the TR will be poor because such an assumption is incorrect. Condition 4 is typically difficult to assess and estimates of the cross-correlation parameters depend heavily on the design. This motivates the introduction of a new class of designs in Section 3. What follows is a heuristic example to illustrate the fact that Conditions 1 and 2 are not essential, but Conditions 3 and 4 are important. This example also demonstrates that when Condition 3 or 4 are not satisfied, the individual kriging (IK) model is more appropriate. Unlike the integral kriging models, the IK model allows different values of μ, σ^2, and $\boldsymbol{\theta}$ for different responses.

Example 1. Consider an experiment involving one qualitative variable of two levels, denoted by 1 and 2, and one quantitative variable x. The simulated response curves are $y(1, x) = a_1 \sin(b_1 * \pi * (x + 1/8c_1)) + d_1$ (Curve I) and $y(2, x) = a_2 \sin(b_2 * \pi * (x + 1/8c_2)) + d_2$ (Curve II) with $x \in [0, 1)$. For each curve, the coefficients are drawn from independent Gaussian distributions with standard deviation 0.01. For each level of the qualitative variable, the training data are generated by using a random Latin hypercube design (McKay, Beckman, and Conover 1979) with six runs for x in $[0, 1)$, and the testing data are taken at 100 equally spaced points in $[0, 1)$. The data for each response curve are normalized before building the models. The root-mean-squared error (RMSE) of the predictor (5) over N_0 testing points $\mathbf{w}_1^*, \mathbf{w}_2^*, \ldots, \mathbf{w}_{N_0}^*$, defined as $\text{RMSE} = N_0^{-1}\sqrt{\sum_{u=1}^{N_0}(y(\mathbf{w}_u^*) - \hat{y}(\mathbf{w}_u^*))^2}$, is used to assess the prediction accuracy. This procedure of data generation, modeling, and prediction accuracy assessment is repeated 100 times for each integral kriging model and the IK model. Four scenarios are considered below, which are chosen to represent departures corresponding to Conditions 1–4, respectively.

Scenario	Case 1	Case 2
1	(1, 4, 0.125, 0)	(1, 4, 0.125, 0 to 5 by 1.0)
2	(1, 4, 0.125, 0)	(1 to 6 by 1.0, 4, 0.125, 0)
3	(1, 4, 0.125, 0)	(1, 4 to 5 by 0.2, 0.125, 0)
4	(1, 4, 0.125, 0)	(1, 4, 0 to 2 by 0.2, 0)

NOTE: • The column "Case 1" contains the mean values of (a_1, b_1, c_1, d_1) for each scenario. • The column "Case 2" contains the mean values of (a_2, b_2, c_2, d_2) for each scenario. • Scenarios 1–4 are chosen to represent departures corresponding to Conditions 1–4, respectively.

Figure 1 shows the prediction performances of the IK and integral kriging models under Scenarios 1–4 respectively, when Curve I is set to be the TR. The situations for predicting Curve II are similar so we omit them for saving space.

The first row of Figure 1 demonstrates the nonessential nature of Condition 1 when Conditions 2–4 hold. As the difference in the overall means between the two curves increases, the medians of the RMSEs of the integral kriging models change little and they are always smaller than the medians of the RMSEs of the IK model, and their standard deviations are comparable. The second row presents a similar phenomenon, which demonstrates the nonessential nature of Condition 2. The third row demonstrates that as the difference in the frequencies of the oscillation increases, prediction performances of the integral kriging models deteriorate and will be inferior to the IK model when the difference is sufficiently large. The fourth row demonstrates a similar phenomenon to the third row as the difference in the phases of the oscillation increases (which means that the cross-correlation between the curves is weakening). Note that prediction performance of the UC model takes a favorable turn as b_2 varies from 1 to 2. This benefit derives from the fact that the cross-correlation parameters of the UC model can take values in $(-1, 1)$, whereas the cross-correlation parameters of the EC and MC models can only take values in $(0, 1)$.

At this point, it should be clear that the verification of Conditions 3 and 4 is crucial for building the integral kriging models. In fact, Condition 3 is a necessary condition of Condition 4, so it suffices to verify Condition 4. For brevity, we call an AR satisfying Condition 4 the one similar to the TR. A natural question arises on how to measure the similarities between the AR and the TR. To tackle this problem, a data-driven method from an experimental design viewpoint will be proposed in the next section.

3. OPTIMAL CSLHD AND ITS CONSTRUCTION

From the discussions in Section 2, two responses are said to be similar if their phases of oscillation are nearly the same or opposite. To measure the similarities among different responses, a natural approach is to use the *sample correlation coefficients* among the output vectors of different responses as the similarity measures. From the modeling perspective, using the same design helps to yield a large (in absolute value) cross-correlation for a pair of responses that are similar. On the other hand, using different designs for the responses helps to provide information from a wider sample of the quantitative variables. These two objectives are competing with each other. To obtain a large cross-correlation between two similar responses and make good

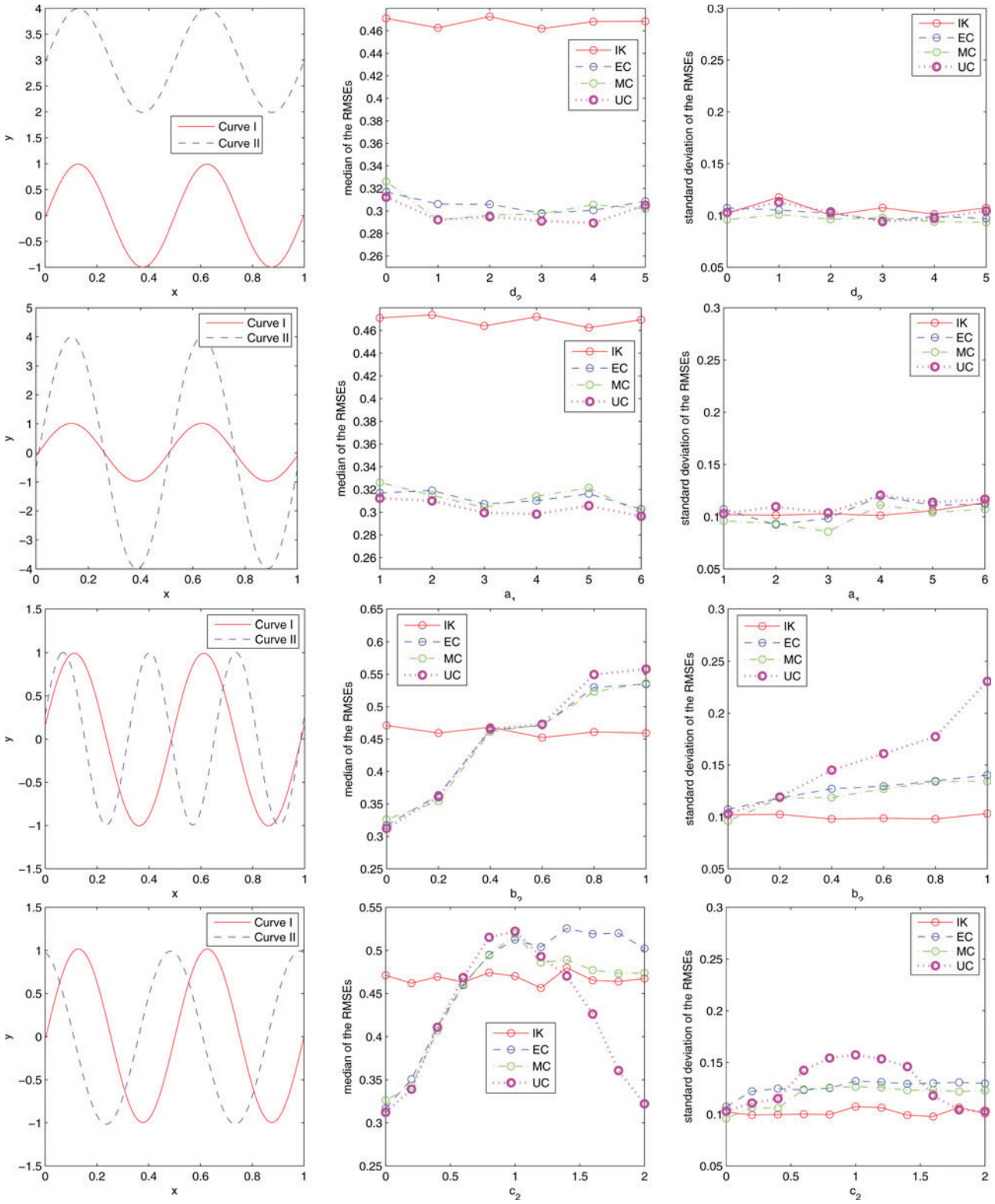

Figure 1. Rows 1–4 represent Scenarios 1–4, respectively. For each row, the left panel shows the plot of the curves for one draw ($d_2 = 3$, $a_2 = 4$, $b_2 = 4.6$, $c_2 = 1$ for rows 1–4, respectively); the middle panel shows the medians of the RMSEs when Curve I is set to be the TR; the right panel shows the standard deviations of the RMSEs.

use of information from the AR, designs that balance the two competing objectives are desirable. The objective of obtaining a large cross-correlation between two similar responses is more important since it guarantees that the information of the AR can be used effectively. To this end, making the designs similar for each response but not identical helps meet the first objective and also provides information from a wide sample of the quantitative variables. A simple approach is to use the same design for all responses, then add a small amount of random jitter to each point. This approach, however, is not guaranteed to achieve

good stratification for each dimension, which makes it difficult to use information from the quantitative variables. A systematic method to locate the design points is called for.

In this section, a new design called clustered-sliced Latin hypercube design (CSLHD) is proposed first. This design is useful for measuring the similarities among different responses, while keeping most of the desirable properties of an SLHD. A computer algorithm for constructing optimal CSLHDs (OCSLHDs) under the centered L_2-discrepancy (Hickernell 1998) will then be developed. The similarity measures, produced by the OCSLHDs, will be used to further determine which ARs are indeed useful. This issue will be elaborated in Section 4.

3.1 Clustered-Sliced Latin Hypercube Designs

Let n be the number of design points for each level-combination of the qualitative variables, s be the number of level-combinations of the qualitative variables, and $N = ns$. Assume that the design region for the quantitative variables is $[0, 1)^t$, and the quantitative variables can be varied independently. A CSLHD can be constructed via the following four steps:

Algorithm 1.

Step 1. For $i = 1, \ldots, n$, let $\mathbf{g}_i = ((i-1)s+1, (i-1)s+2, \ldots, is)^T$.

Step 2. Let $(u_1, \ldots, u_n)^T$ be a permutation on $(1, \ldots, n)^T$. For $i = 1, \ldots, n$, let $\mathbf{g}_i^\star$ be a uniform permutation on $\mathbf{g}_{u_i}$ (each possible permutation will be selected with an equal probability), and $\mathbf{h}_k = (\mathbf{g}_1^\star(k), \ldots, \mathbf{g}_n^\star(k))^T$, for $k = 1, \ldots, s$. Put $\mathbf{h} = (\mathbf{h}_1^T, \ldots, \mathbf{h}_s^T)^T$, which is an $N \times 1$ column vector.

Step 3. Repeat Step 2 t times independently; each time an $N \times 1$ column vector $\mathbf{h}$ is generated, and juxtapose those t column vectors, column by column; an $N \times t$ array $\mathbf{G}$ is obtained.

Step 4. The resulting design $\mathbf{D}$ is generated by $\mathbf{D} = (\mathbf{G} - \mathbf{J}_{N,t})/N$, where $\mathbf{J}_{N,t}$ is an $N \times t$ matrix of $(1/2)$'s or independent deviates all having the uniform distribution on $(0, 1)$.

The general properties for the proposed design are summarized as below, with proofs given in Supplementary Section S2.

Proposition 1. For the design $\mathbf{D}$ constructed via Algorithm 1, we have that

(i) $\mathbf{D}$ is a Latin hypercube design with N levels;

(ii) For $k = 1, \ldots, s$, let $\mathbf{D}^{(k)}$ be the kth slice consisting of rows $(k-1)n+1, \ldots, kn$ of $\mathbf{D}$, then $\mathbf{D}^{(k)}$ is a Latin hypercube design with n levels;

(iii) Let $\mathbf{J}_{N,t}$ be an $N \times t$ matrix of $(1/2)$'s. For $i = 1, \ldots, n$, let $\mathbf{A}^{(i)}$ be the ith subarray of $\mathbf{D}$ consisting of all the ith rows of $\mathbf{D}^{(1)}, \ldots, \mathbf{D}^{(s)}$, and let M_i be the maximum interpoint distance for $\mathbf{A}^{(i)}$. We have $M = \max_{1 \le i \le n} M_i \le (1 - s^{-1})\sqrt{t}/n$.

Proposition 1 (i) and (ii) guarantee that the constructed design is an SLHD. It is a new class of SLHD with N runs, t columns, and s slices of equal size. Such a design is denoted by CSLHD(N, t, s). A small value of the M defined in Proposition 1 (iii) indicates that the design points of the proposed design have a clustered structure, that is, as pointed out by one referee, for any point in one slice there will be one point in another slice within a distance M. Thus, a small M implies that the slices of a CSLHD are nearly the same. This is a desirable property that ensures the sample correlation coefficients among the output vectors can be used as the similarity measures. Note that if the unit cube is divided into N^t cells of the same size, we allow the points of a CSLHD(N, t, s) located at the centers of the N corresponding cells or selected at random from them (selection of $\mathbf{J}_{N,t}$ in Step 4). Selecting points from the centers of cells facilitates the construction of optimal designs (see Section 3.2), whereas selecting at random from the cells gives the design similar sampling properties as an SLHD (Qian 2012).

The following example illustrates the construction steps of a CSLHD. Another example is provided in Supplementary Section S3 to demonstrate the application of CSLHDs in a numerical integration problem. The example below has the points selected from the centers of cells, whereas the supplementary example has the points selected at random from cells.

Example 2. Let $n = 6, t = 2, s = 3$ and $N = ns = 18$. Our construction method will give a CSLHD$(18, 2, 3)$ in the four steps as described in Algorithm 1.

Step 1. Let $\mathbf{g}_1 = (1, 2, 3)^T$, $\mathbf{g}_2 = (4, 5, 6)^T$, $\mathbf{g}_3 = (7, 8, 9)^T$, $\mathbf{g}_4 = (10, 11, 12)^T$, $\mathbf{g}_5 = (13, 14, 15)^T$ and $\mathbf{g}_6 = (16, 17, 18)^T$.

Step 2. Take $(4, 2, 5, 6, 3, 1)^T$ as one permutation on $(1, 2, 3, 4, 5, 6)^T$. Then, permutations on $\mathbf{g}_4, \mathbf{g}_2, \mathbf{g}_5, \mathbf{g}_6, \mathbf{g}_3$ and $\mathbf{g}_1$ result in $\mathbf{g}_1^\star = (12, 11, 10)^T$, $\mathbf{g}_2^\star = (4, 5, 6)^T$, $\mathbf{g}_3^\star = (14, 15, 13)^T$, $\mathbf{g}_4^\star = (18, 17, 16)^T$, $\mathbf{g}_5^\star = (8, 9, 7)^T$ and $\mathbf{g}_6^{xyz} = (1, 3, 2)^T$, respectively. Thus, the first column of $\mathbf{G}$ is $\mathbf{h} = (\mathbf{h}_1^T, \mathbf{h}_2^T, \mathbf{h}_3^T)^T$, where $\mathbf{h}_1 = (12, 4, 14, 18, 8, 1)^T$, $\mathbf{h}_2 = (11, 5, 15, 17, 9, 3)^T$ and $\mathbf{h}_3 = (10, 6, 13, 16, 7, 2)^T$.

Step 3. Repeat the above procedure and obtain the second column of $\mathbf{G}$ as $(2, 14, 7, 11, 4, 18, 3, 13, 9, 12, 6, 17, 1, 15, 8, 10, 5, 16)^T$.

Step 4. The resulting array $\mathbf{G}$ and the corresponding design $\mathbf{D}$ in $[0, 1)^2$ can be obtained as shown in Figure 2. The details are given in Supplementary Section S4.

The left panel of Figure 2 presents the bivariate projections of $\mathbf{D}$, where each of the 18 equally spaced intervals in $[0, 1)$ contains precisely one point (i.e., it is an LHD). The points marked with "$\circ$" come from the first slice $\mathbf{D}^{(1)}$, the points marked with "$+$" come from the second slice $\mathbf{D}^{(2)}$, and the points marked with "$*$" come from the third slice $\mathbf{D}^{(3)}$. It is easy to see that the three slices are all LHDs. Furthermore, from Proposition 1 (iii), we have

$$\mathbf{A}^{(1)} = \begin{pmatrix} 0.6389 & 0.0833 \\ 0.5833 & 0.1389 \\ 0.5278 & 0.0278 \end{pmatrix}, \quad \mathbf{A}^{(2)} = \begin{pmatrix} 0.1944 & 0.7500 \\ 0.2500 & 0.6944 \\ 0.3056 & 0.8056 \end{pmatrix},$$

$$\mathbf{A}^{(3)} = \begin{pmatrix} 0.7500 & 0.3611 \\ 0.8056 & 0.4722 \\ 0.6944 & 0.4167 \end{pmatrix}, \quad \mathbf{A}^{(4)} = \begin{pmatrix} 0.9722 & 0.5833 \\ 0.9167 & 0.6389 \\ 0.8611 & 0.5278 \end{pmatrix},$$

$$\mathbf{A}^{(5)} = \begin{pmatrix} 0.4167 & 0.1944 \\ 0.4722 & 0.3056 \\ 0.3611 & 0.2500 \end{pmatrix}, \quad \mathbf{A}^{(6)} = \begin{pmatrix} 0.0278 & 0.9722 \\ 0.1389 & 0.9167 \\ 0.0833 & 0.8611 \end{pmatrix}.$$

For $i = 1, \ldots, 6$, all three points of $\mathbf{A}^{(i)}$ fall into the same 3×3 square and are close to each other. This shows the clustered structure of the constructed design. □

Remark 1. As pointed out by one referee, the CSLHD constructed by Algorithm 1 is similar to the cascading Latin hypercube design (Handcock 1991). An alternative construction method for the CSLHDs is provided in Supplementary Section S5. The resulting design from this alternative construction method also has the properties presented in Proposition 1 via a proper permutation of the runs.

For $i = 1, \ldots, s$, let $\mathbf{y}^{(i)} = (\mathbf{y}^{(i)}(1), \ldots, \mathbf{y}^{(i)}(n))^T$ be the output vector corresponding to $\mathbf{D}^{(i)}$, where $\mathbf{D}^{(i)}$ is the design for the quantitative variables corresponding to the ith level-combination of the qualitative variables. If the value of the M defined in Proposition 1 (iii) is small, $\mathbf{D}^{(i)}$'s are nearly the same. On the other hand, the whole design will keep the space-filling properties of an SLHD, that is, univariate uniformity for each slice as well as for the whole design. Hence, the sample correlation coefficients among $\mathbf{y}^{(i)}$'s can be used as the similarity measures among different responses. By Proposition 1 (iii), a large n usually means a small bound of M. Moreover, the upper bound of M is seldom achieved due to the harsh condition for the equality. The upper bound of M could be user-defined. Based upon our empirical observations through numerous simulations, $M \leq 0.15$ is recommended as a standard choice (e.g., $M = 0.12$ in Example 2). Thus for any given t and s, only the value of n needs to be determined, such that $(1 - s^{-1})\sqrt{t}/n \leq 0.15$. A CSLHD$(N, t, s)$ with $N = ns$ can always be constructed by Algorithm 1 for such a specified n.

An analysis strategy based on the sample correlation coefficients (similarity measures) among $\mathbf{y}^{(i)}$'s will be developed in Section 4 to further determine which ARs are really useful. Could the cross-correlation parameters in (3) or (4) be used as the similarity measures among the responses? Note that this is reasonable only if the model assumptions in Section 2 are satisfied. Furthermore, the PDUDE property does not allow a cross-correlation parameter to be -1. So if two responses are completely negatively correlated with each other, the cross-correlation would not be sufficiently accurate. For these reasons, the cross-correlation parameters in (3) or (4) will not be considered as the measures of similarity in this work.

3.2 Construction of Optimal CSLHDs

A CSLHD guarantees the univariate uniformity in each slice, but a good space-filling property for high-dimensional projections is also desirable. This section will focus on this issue. The basic idea is to construct the optimal CSLHD (OCSLHD) based on some proper design criterion. Practitioners would use the OCSLHDs rather than the CSLHDs constructed via Algorithm 1. However, Algorithm 1 provides the foundation for the construction of the OCSLHDs. To facilitate the design construction, $\mathbf{J}_{N,t}$ in *Step 4* of Algorithm 1 is taken to be the $N \times t$ matrix of $(1/2)$'s.

The design criterion used here is the centered L_2-discrepancy (CL_2) proposed by Hickernell (1998). The closed form for calculating the CL_2 value of a design $\mathbf{D} = (d_{ij})$ with N runs and t factors in $[0, 1)^t$, denoted by $CL_2(\mathbf{D})$, can be found in the Supplementary Section S6. The CL_2 is considered here due to its invariance and flexible projection properties. In addition, the CL_2 is a well-known space-filling criterion, and extensive empirical studies have revealed that space-filling designs are suitable for the GP model (see Santner, Williams, and Notz 2003; Fang, Li, and Sudjianto 2006). Other criteria, such as entropy (Shewry and Wynn 1987), minimax and maximin distance (Johnson, Moore, and Ylvisaker 1990), and various discrepancies (Fang, Li, and Sudjianto 2006) can be used as well for selecting the OCSLHDs.

Let $\mathcal{D}$ be the class of CSLHDs of N runs, t columns, and s slices of n runs each. Then the optimization problem for the OCSLHD is to find a CSLHD $\mathbf{D}^* \in \mathcal{D}$ such that $CL_2(\mathbf{D}^*) = \min_{\mathbf{D} \in \mathcal{D}} CL_2(\mathbf{D})$. We next define the neighborhood of a CSLHD $\mathbf{D}^c$ formed by all nearby designs of $\mathbf{D}^c$. The formulation of the neighborhood of $\mathbf{D}^c$ is summarized in five steps.

Algorithm 2.

Step 1. Randomly select one column from $\mathbf{D}^c$ and write the $N \times 1$ column vector as $\mathbf{d} = (\mathbf{d}_{(1)}^T, \ldots, \mathbf{d}_{(s)}^T)^T$, where $\mathbf{d}_{(i)} = (d_1^{(i)}, \ldots, d_n^{(i)})^T$, $i = 1, \ldots, s$. Let $\mathbf{E}_1 = (\mathbf{d}_{(1)}, \ldots, \mathbf{d}_{(s)})$ be the $n \times s$ matrix obtained by combining $\mathbf{d}_{(i)}$'s column by column.

Step 2. Randomly choose u and v such that $1 \leq u < v \leq n$. Let $\mathbf{E}_2 = (e_{ij})_{n \times s}$ be the matrix obtained by exchanging the uth and vth rows of $\mathbf{E}_1$.

Step 3. Randomly choose i_1 and i_2 such that $1 \leq i_1 < i_2 \leq s$. Let $\mathbf{F} = (f_{ij})_{n \times s}$ be the matrix obtained by exchanging the (u, i_1)th and (u, i_2)th entries (or the (v, i_1)th and (v, i_2)th entries) of $\mathbf{E}_2$.

Step 4. Let $\mathbf{f} = (\mathbf{f}_{(1)}^T, \ldots, \mathbf{f}_{(s)}^T)^T$ with $\mathbf{f}_{(i)}$ being the ith column of $\mathbf{F}$. Replace the column $\mathbf{d}$ in $\mathbf{D}^c$ by $\mathbf{f}$ to obtain a new design $\mathbf{D}^{\text{new}}$.

Step 5. All $\mathbf{D}^{\text{new}}$'s created by Steps 1–4 form the neighborhood of $\mathbf{D}^c$.

This formulation of neighborhood is known as the column-exchange approach (Li and Wu 1997; Ye, Li, and Sudjianto 2000; Fang, Li, and Sudjianto 2006). This approach is a popular choice in the literature as it maintains the structure of the design. For example, Morris and Mitchell (1995) used this approach as a basis to construct optimal LHDs; Ye, Li, and Sudjianto (2000) used it for constructing optimal symmetric LHDs; Fang et al. (2000) used it for constructing uniform designs.

Based on the neighborhood constructed by Algorithm 2, a simulated annealing algorithm implemented in matlab, denoted by ALA(T_0, α, N_T, M_0), is adopted for constructing the OCSLHDs, where T_0 is the initial temperature, α is a cooling parameter, N_T is the number of designs constructed at each temperature, and M_0 is the total number of temperature changes (see Fang, Li, and Sudjianto 2006 for details). The simulated annealing algorithm is a powerful optimization algorithm when the objective function has many local optima (which often occurs when constructing optimal designs). This algorithm moves from $\mathbf{D}^c$ to $\mathbf{D}^{\text{new}}$ with a replacement probability even $\mathbf{D}^{\text{new}}$ is inferior to $\mathbf{D}^c$, thus has chances to escape from a local optimum. Although the convergence rate of the simulated annealing is not fast, it is easy-to-apply and has been successfully implemented;

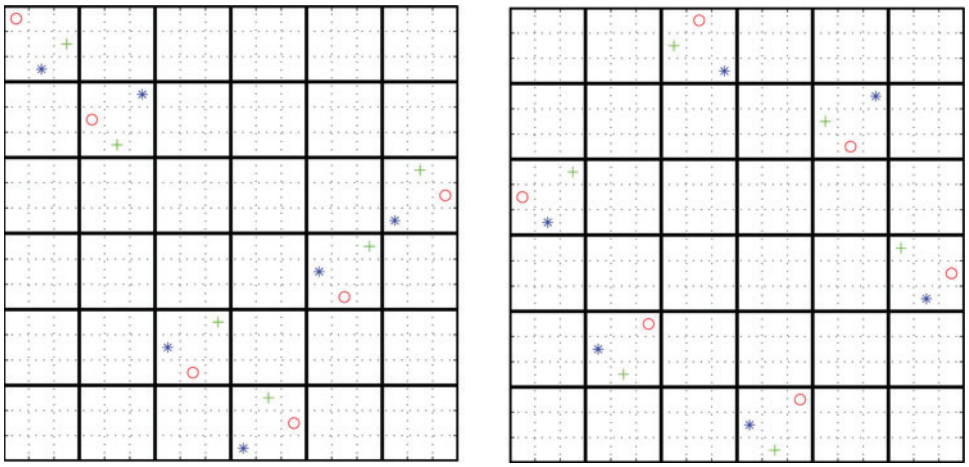

Figure 2. Bivariate projections of the CSLHD(18,2,3) in Example 2 (left panel) and the OCSLHD(18,2,3) in Example 3 (right panel).

for example, in the work of Morris and Mitchell (1995), and Qian et al. (2006).

Example 3. Take $\mathbf{D}^{\mathrm{c}}$ as the CSLHD(18, 2, 3) in the left panel of Figure 2. One nearby design of $\mathbf{D}^{\mathrm{c}}$ can be generated by the first four steps of Algorithm 2, whose details are given in Supplementary Section S7. Using $\mathbf{D}^{\mathrm{c}}$ as the initial design, the algorithm ALA(100, 0.5, 30, 15) results in the OCSLHD(18, 2, 3), denoted as $\mathbf{D}^{\mathrm{o}}$. The CPU time for searching $\mathbf{D}^{\mathrm{o}}$ is about 20 s on a triple-core 2.4-GHz PC. The bivariate projection of $\mathbf{D}^{\mathrm{o}}$ is shown in the right panel of Figure 2, and $CL_2(\mathbf{D}^{\mathrm{c}}) = 0.0060$ and $CL_2(\mathbf{D}^{\mathrm{o}}) = 0.0019$. This shows that $\mathbf{D}^{\mathrm{o}}$ significantly improves the space-filling property of $\mathbf{D}^{\mathrm{c}}$, as can be seen from the two panels of Figure 2.

4. SELECTION PROCEDURE

For the OCSLHDs constructed in Section 3.2, the sample correlation coefficients among the output vectors of different slices can be used as the similarity measures among the responses. However, these measures do not tell whether the corresponding ARs are really useful for predicting the TR. A way is required to make a further judgment. This section develops a forward selection procedure which uses the similarity measures produced by the OCSLHDs.

The basic idea of the procedure is to sequentially add the slices (and the corresponding output vectors) according to some criterion. The slices are included as additional rows, not as additional columns, that is, the ARs are treated as additional data rather than additional variables. Specifically, ARs with larger magnitude similarity measures are more likely to be included for modeling. In other words, the magnitudes of the similarity measures provide an entering order for the selection procedure. In the process of the selection, we begin with the data of the TR and sequentially add the data of the ARs as additional rows, following the entering order determined by the magnitudes of the similarity measures. An important issue is when we should stop adding data. This is addressed by using the leave-one-out cross-validation (CV) approach which is a commonly used technique in computer experiments to assess the accuracy of the metamodels (see Fang, Li, and Sudjianto 2006; Qian, Wu, and Wu 2008; Han et al. 2010).

For the $N \times (t + l)$ design matrix $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_N)^T$ and the $N \times 1$ response vector $\mathbf{y} = (y_1, \ldots, y_N)^T$, the leave-one-out CV score is defined as

$$\psi = \frac{1}{N}\sum_{i=1}^{N}\{y_i - \hat{y}_{-i}(\mathbf{w}_i)\}^2, \tag{6}$$

where $\hat{y}_{-i}(\mathbf{w}_i)$ is the BLUP (see (5)) at $\mathbf{w}_i$ obtained from the metamodel based on the samples excluding $(\mathbf{w}_i, y_i)$.

Of course, the CV score can be calculated based on any subarray of $\mathbf{W}$ and the corresponding subvector of $\mathbf{y}$. For $i = 1, \ldots, s$, let $\mathbf{W}^{(i)}$ and $\mathbf{y}^{(i)}$ denote the subarray of $\mathbf{W}$ and subvector of $\mathbf{y}$ corresponding to the ith level-combination of the qualitative variables. For brevity, the response corresponding to the ith level-combination of the qualitative variables is called the ith response. In summary, the proposed forward selection procedure works as follows: if the ith response is set to be the TR, add the data of the ARs one by one following the entering order determined by the similarity measures until the CV score increases—an evidence that the accuracy of the model decreases. A detailed algorithm is stated as follows.

Algorithm 3.

Step 1. Calculate the CV score $\psi^{(i)}$ in (6) based on $\mathbf{W}^{(i)}$ and $\mathbf{y}^{(i)}$ using the IK model. Set $\mathbf{W}^{\mathrm{c}} = \mathbf{W}^{(i)}$, $\mathbf{y}^{\mathrm{c}} = \mathbf{y}^{(i)}$, $\psi^{\mathrm{c}} = \psi^{(i)}$.

Step 2. Set $\mathcal{I} = \{j_1, j_2, \ldots, j_{s-1}\} = \{1, \ldots, s\} \setminus \{i\}$ with elements being sorted such that the absolute correlation coefficients between $\mathbf{y}^{(j_k)}$ and $\mathbf{y}^{(i)}$ for $k = 1, \ldots, s-1$ are ranked in descending order, and $\mathcal{J} = \{i\}$.

Step 3. If $\mathcal{I}$ is empty, output $\mathcal{J}$ and the procedure stops; otherwise, let j^* be the first element of $\mathcal{I}$, calculate the CV score $\psi^{(j^*)}$ based on $((\mathbf{W}^{\mathrm{c}})^T, (\mathbf{W}^{(j^*)})^T)^T$ and $((\mathbf{y}^{\mathrm{c}})^T, (\mathbf{y}^{(j^*)})^T)^T$ using one of the integral kriging models.

Step 4. If $\psi^{(j^*)} \le \psi^{\mathrm{c}}$, set $\mathcal{I} = \mathcal{I} \setminus \{j^*\}$, $\mathcal{J} = \mathcal{J} \cup \{j^*\}$, $\psi^{\mathrm{c}} = \psi^{(j^*)}$, $\mathbf{W}^{\mathrm{c}} = ((\mathbf{W}^{\mathrm{c}})^T, (\mathbf{W}^{(j^*)})^T)^T$ and $\mathbf{y}^{\mathrm{c}} = ((\mathbf{y}^{\mathrm{c}})^T, (\mathbf{y}^{(j^*)})^T)^T$, go to Step 3; otherwise, output $\mathcal{J}$ and the procedure stops.

Output $\mathcal{J}$ by Algorithm 3 identifies which ARs are useful. The responses indexed by $\mathcal{J}$ are expected to be similar to the TR, and will be used for predicting the TR. Specifically, an IK model will be built if $\mathcal{J}$ contains only one element; otherwise, the integral kriging models shall be built based on all information of the ARs indexed by $\mathcal{J}$. As a matter of fact, Algorithm 3 can be viewed

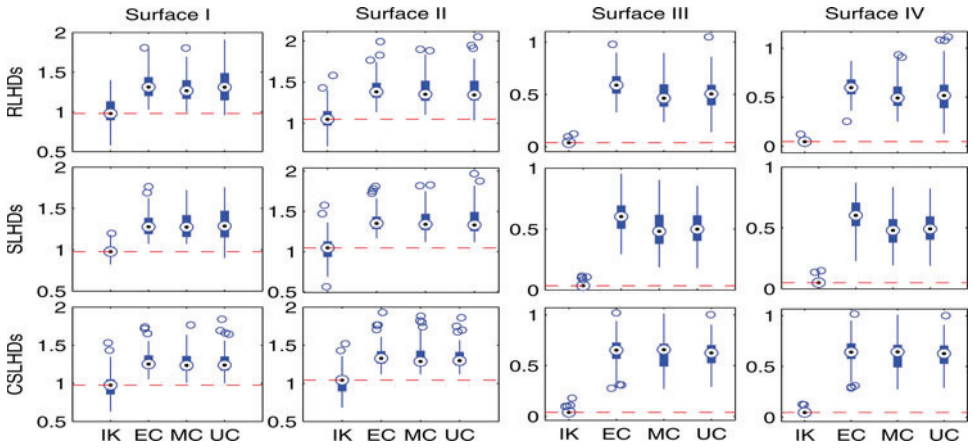

Figure 3. Boxplots of the RMSEs of the three random designs without any selection procedure.

as an estimation procedure if Conditions 1–3 in Section 2 are satisfied. That is, the cross-correlation values among the TR and the useful ARs are believed to be reasonably large (in absolute value), and those among the TR and the useless ARs are believed to be close to zero (and thus set to be zero). The consistency of the CV approach under some general conditions had been proved by Stone (1977). Thus, it is strongly recommended that n should be as large as economically feasible, not only to provide good modeling and prediction, but also to justify the rationality of Algorithm 3.

Remark 2. Using Algorithm 3, the OCSLHDs are effective for determining a sensible entering order of the ARs. Other design types may not work so well because meaningful similarity measures would likely be missed. Moreover, Algorithm 3 results in a design that has an equal number of runs and a good space-filling property for each response.

Remark 3. Algorithm 3 is also suitable when using an identical design (ID) with a good space-filling property for all the level-combinations of the qualitative variables. This is because the sample correlation coefficients among the output vectors produced by a space-filling ID can also be used as the similarity measures among different responses. As discussed at the beginning of Section 3, however, its poor projection property for the quantitative variables may be not beneficial to prediction. Moreover, as pointed out by one referee, if the output is not sensitive to some level-combinations of the qualitative variables, these level-combinations can be set to be a nominal level. In this case, duplicate runs may result, which is not desirable in deterministic computer experiments.

5. SIMULATION EXAMPLES

This section presents two examples to investigate the effectiveness of the interface between the OCSLHDs constructed in Section 3.2 and the selection procedure developed in Section 4. Performances of several commonly used design types are also investigated for the comparison. The first example has one qualitative variable and one quantitative variable. The second example has two qualitative variables and one quantitative variable. Both of these two examples assume that there are some similar responses but not all the responses are similar, which is believed to be common in practice. Therefore, the OCSLHDs are expected to produce promising results when using the proposed selection procedure. One more example (one qualitative variable and five quantitative variables) is provided in Supplementary Section S8. We have also applied the OCSLHDs associated with the proposed selection procedure to several other simulated examples having a single or multiple qualitative or quantitative variables, and similar successes have been obtained.

Example 4. Consider an experiment with two quantitative variables $(x_1, x_2) \in [0, 1)^2$, and one qualitative variable z of four levels. The correlation function (3) is adopted for the integral kriging models. The true response surfaces of this experiment are: $y(1, x_1, x_2) = a_1 \cos(\theta_1 \pi (x_1 + x_2))$ (Surface I); $y(2, x_1, x_2) = b_1 \cos(\theta_2 \pi (x_1 + x_2)) + b_2$ (Surface II); $y(3, x_1, x_2) = c_1 (x_1 + x_2 - c_2)^2 + c_3 \sin(\theta_3 \pi (x_1 + x_2)) + c_4$ (Surface III); and $y(4, x_1, x_2) = d_1 (x_1 + x_2 - d_2)^2 + d_3 \sin(\theta_4 \pi (x_1 + x_2)) + d_4$(Surface IV). For each response surface, the coefficients are drawn from independent Gaussian distributions with standard deviation 10^{-2}. The mean values of (a_1, θ_1), (b_1, b_2, θ_2), $(c_1, c_2, c_3, c_4, \theta_3)$, and $(d_1, d_2, d_3, d_4, \theta_4)$ are set to be $(2, 3.5)$, $(-2, -4, 3.5)$, $(20, 1, 0.1, 3, 2)$, and $(30, 1, 0.1, 8, 2)$, respectively. The true response surfaces for one draw is displayed in Figure 3 of Supplementary Section S9.

From the closed forms of Surfaces I—IV (or Figure 3 of Supplementary Section S9), it is clear that Surfaces I and II are similar, and Surfaces III and IV are similar as well. Sixteen training points are generated for each surface by using six design types, that is, (1) random Latin hypercube designs (RLHDs) from McKay, Beckman, and Conover (1979); (2) SLHDs from Qian (2012); (3) CSLHDs constructed in Section 3.1; (4) the uniform design (IUD) available at *http://sites.stat.psu.edu/~rli/DMCE/UniformDesign/*; (5) the uniform SLHD (USLHD) from Chen et al. (2014); (6) the OCSLHD constructed in Section 3.2. The latter three are the optimal versions of the former three under the CL_2 criterion, respectively. The IUD takes a uniform design for one response, then replicates it exactly for the other responses.

In this example, the upper bound of M defined in Proposition 1 is 0.07 for CSLHDs and the OCSLHD. The CPU time for gen-

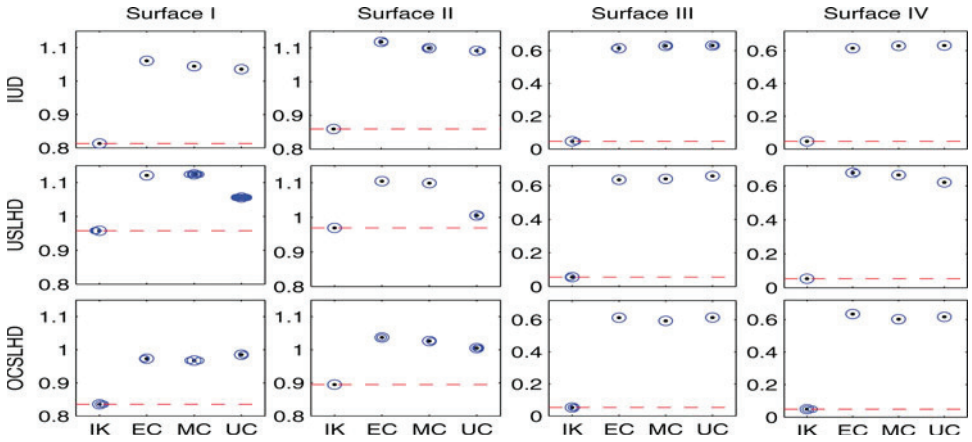

Figure 4. Boxplots of the RMSEs of the three optimal designs without any selection procedure.

erating the OCSLHD(64, 2, 4) is about 35s on a triple-core 2.4-GHz PC. The testing data are taken on a grid of $40^2 = 1600$ equispaced points on $[0, 1)^2$ for each response. This procedure of data generation, modeling, and prediction accuracy assessment is repeated 500 times for each integral kriging model and the IK model. For each repetition, the data for each response surface are normalized.

We first investigate the prediction performances when the selection procedure is absent. Figure 3 displays boxplots of the RMSEs of the three random designs, whereas Figure 4 displays the boxplots of the RMSEs of the three optimal designs. Next, we investigate the prediction performances when the selection procedure is present. The CPU time for finishing the selection procedure once is about 16s on a triple-core 2.4-GHz PC. Figure 5 displays the boxplots of the RMSEs of the random designs, whereas Figure 6 displays the boxplots of the RMSEs of the optimal designs. The detailed numerical results under various cases when the selection procedure is present are given in Supplementary Section S12.

The following observations are summarized from this study.

(i) When the proposed selection procedure is absent (Figures 3 and 4), all the integral kriging models work much worse than the IK model across all design types. This is expected since not all of the four true responses are similar.

(ii) When the proposed selection procedure is used (Figures 5 and 6), the prediction performances of the integral kriging models are at least comparable with those of the IK model across all design types. Among the random designs, the CSLHDs typically outperform the RLHDs and SLHDs. A similar phenomenon occurs among the optimal designs, that is, the OCSLHD typically outperforms the IUD and USLHD in the sense that it results in the smallest RMSEs when the integral kriging models are built. Besides, the OCSLHD outperforms the CSLHDs, which is expected since the latter is the optimal version of the former.

(iii) When Surface I or II is set to be the TR, the EC and MC models are not better than the IK model. This is because both the EC and MC models can only capture positive correlations, while Surfaces I and II are clearly negatively correlated since a_1 and b_1 are opposite to each other and $\theta_1 = \theta_2$.

Example 5. Consider an experiment with two qualitative variables both at two levels, and one quantitative variable in $[0, 1)$. The correlation function (4) is adopted for the integral kriging models in this example. The true response curves are generated using the following four equations extracted from Han et al. (2010): $y(1, 1, x) = 0.3 + 0.3x + 0.1$

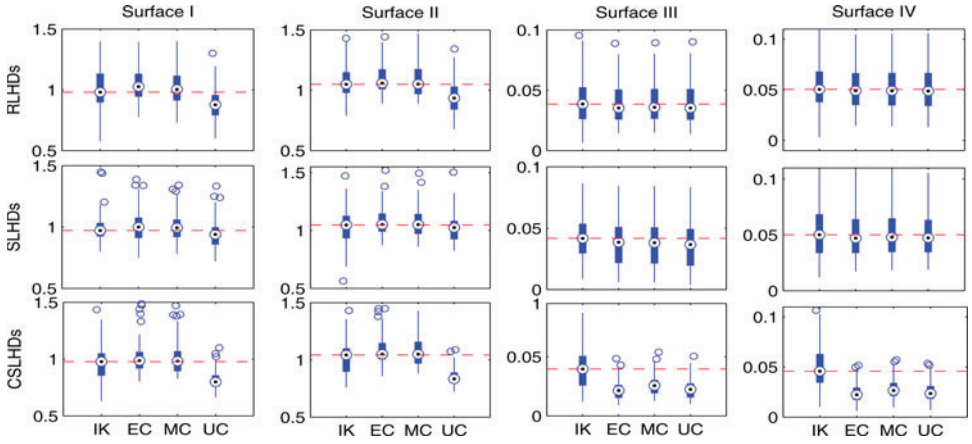

Figure 5. Boxplots of the RMSEs of the random designs with the proposed selection procedure.

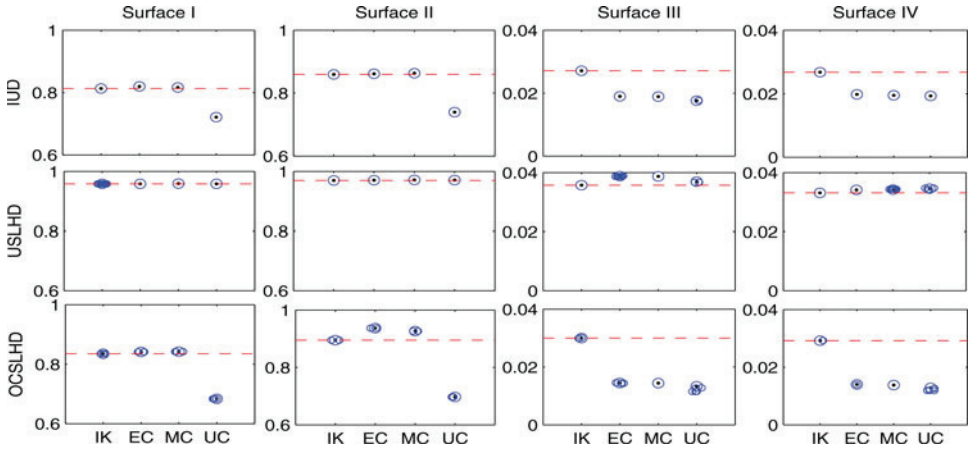

Figure 6. Boxplots of the RMSEs of the optimal designs with the proposed selection procedure.

$\sin(2\pi x) + 2.5(x - 0.5)^4 - 0.4x^5$; $y(2, 1, x) = 0.2 + 0.3x + 0.1\sin(2\pi x) + 0.5(x - 0.5)^2$; $y(1, 2, x) = 0.1 + 0.3x + 0.1\sin(2.5\pi x) + 2.5(x - 0.5)^4 - 0.4x^5$; and $y(2, 2, x) = 0.3x + 0.1\sin(2.5\pi x) + 0.5(x - 0.5)^2$. As in Han et al. (2010), $y(2, 2, x)$ is set to be the TR and only one experiment is conducted. In fact, repeated experiments produce the same numerical results in this example. Since there is only one quantitative variable, the random designs RLHDs, SLHDs, and CSLHDs are exactly the same as the optimal designs IUD, USLHD, and OCSLHD, respectively. So, we only consider the prediction performances of IUD, USLHD, and OCSLHD.

Eight training points are available for each response curve, the data for each response curve are then normalized. The upper bound of M defined in Proposition 1 is 0.09 for the OCSLHD(32, 1, 4). The CPU time for generating the OCSLHD(32, 1, 4) is about 0.5 s on our computer. The RMSEs of IK, EC, MC, and UC with the proposed selection procedure are evaluated at 101 testing points $x \in \{0, 0.01, \ldots, 0.99, 1\}$, and the CPU time for finishing the selection procedure once is about 5.5 s. The numerical results are listed in Table 1.

In Table 1, both the OCSLHD and IUD identify the ARs $y(2, 1, x)$ and $y(1, 2, x)$ for the integral kriging models, the same as in Han et al. (2010). In addition, the OCSLHD has the minimum RMSE values among all design types. This shows that the interface between the OCSLHD and the selection procedure is efficient in terms of the prediction accuracy.

6. A REAL-LIFE EXAMPLE

This section studies a real-life computer experiment to investigate the effectiveness of the interface between the proposed OCSLHD and the selection procedure. A brief description of this case study is given first. For more details, please refer to Dewettinck et al. (1999).

Table 1. Summary of RMSEs for predicting $y(2, 2, x)$ in Example 5

Design type	IK	EC	MC	UC
IUD	0.2622	0.0073(2,3)	0.0067(2,3)	0.0046(2,3)
USLHD	0.2617	0.0074(3)	0.0073(3)	0.0072(3)
OCSLHD	0.2617	0.0007(2,3)	0.0005(2,3)	0.0005(2,3)

NOTE: The data inside the parentheses represent the catched responses, where 2 and 3 stand for $y(2, 1, x)$ and $y(1, 2, x)$, respectively.

Fluidized-bed processes are used in the food industry to coat certain food products. Dewettinck et al. (1999) reported a physical experiment and several associated computer models for predicting the steady-state thermodynamic operation points of a Glatt GPC-1 fluidized-bed unit. The response is affected by six input variables: fluid velocity of the fluidization air (V_f), temperature of the air from the pump (T_a), flow rate of the coating solution (R_f), pressure of atomized air (P_a), temperature (T_r), and humidity (H_r). Dewettinck et al. (1999) conducted a 28-run experiment. For each input setting, one physical response ($T_{2,\text{exp}}$) and three computer responses ($T_{2,1}$, $T_{2,2}$, $T_{2,3}$) were available. There are major differences among the three computer models. Model $T_{2,3}$ is the most accurate (i.e., producing the closest response to $T_{2,\text{exp}}$), model $T_{2,2}$ is the medium accurate, while model $T_{2,1}$ is the least accurate. This is a multi-fidelity computer experiment with four different accuracies. As discussed by Han et al. (2010), a multi-fidelity experiment can also be regarded as an experiment with one qualitative variable whose levels correspond to different accuracies of the response. Thus, this example can be viewed as an experiment with six quantitative variables and one four-level qualitative variable. A fraction of the experimental design (the first seven trials) is presented in Columns 2–7 of Table 2, and the corresponding output results are listed in Columns 8–11 of the same table. The complete experimental design and the output values can be found in Supplementary Section S10.

Four GPs are fitted respectively using the design and the four response vectors in Table 2, then their BLUPs are treated as "true" responses, each of which corresponds to one level of the qualitative variable. We did this so that the effect of different designs and the interfaces between the designs and the proposed selection procedure can be compared. Denote the four BLUPs as $\hat{y}_{2,exp}(1, \mathbf{x})$, $\hat{y}_{2,1}(2, \mathbf{x})$, $\hat{y}_{2,2}(3, \mathbf{x})$, and $\hat{y}_{2,3}(4, \mathbf{x})$, respectively, where $\mathbf{x} = (H_r, T_r, T_a, R_f, P_a, V_f)^T$ and the subscripts indicate the data sources. First, similar to that of Dewettinck et al. (1999), a 28-run experimental design is shared by the four true responses. Note that the original 28-run design should not be considered as it has been used to build the true models. Instead, a 28-run uniform design with six columns is arranged for

Table 2. Settings of input variables and outputs from the physical and computer experiments

Run	$H_r(\%)$	T_r(C)	T_a(C)	R_f(g/min)	P_a(bar)	V_f(m/s)	$T_{2,\exp}$	$T_{2,1}$	$T_{2,2}$	$T_{2,3}$
1	51.00	20.07	50.00	5.52	2.50	3.00	30.40	32.40	31.50	30.20
2	46.40	21.30	60.00	5.53	2.50	3.00	37.60	39.50	38.50	37.00
3	46.60	19.20	70.00	5.53	2.50	3.00	45.10	46.80	45.50	43.70
4	53.10	21.10	80.00	5.51	2.50	3.00	50.20	53.80	52.60	51.00
5	52.00	20.40	90.00	5.21	2.50	3.00	57.90	61.70	59.90	58.20
6	45.60	21.40	60.00	7.25	2.50	3.00	32.90	35.20	34.60	32.60
7	47.30	19.50	70.00	7.23	2.50	3.00	39.50	42.40	41.00	39.10

NOTE: This is a fraction of the data, the complete data can be found in the supplementary material.

each true response to generate the response data. For brevity, we denote this design scheme as the IUD and call the corresponding data as the IUD data. The second design scheme is the proposed OCSLHD($28 \times 4, 6, 4$). That is, one slice of the OCSLHD($28 \times 4, 6, 4$) is designated for one true response, and by this way the experimental data, which we call the OCSLHD data, can be generated. The TR is set to be $\hat{y}_{2,\exp}(1, \mathbf{x})$ which is of the most interest to predict.

After normalizing the data for each response, the proposed selection procedure is carried out for both the IUD and OCSLHD. For each of the EC, MC, and UC models, the proposed selection procedure chooses only $\hat{y}_{2,3}(4, \mathbf{x})$ as the AR. This is because once $\hat{y}_{2,3}(4, \mathbf{x})$ is included, adding the other ARs actually deceases the CV scores although their response vectors appear to be highly correlated with the TR response vector. As described previously, the response $\hat{y}_{2,3}(4, \mathbf{x})$ is established using the most accurate codes. For a comparison purpose, we also use the IUD data to fit the IK, EC, MC, and UC models without any selection procedure. The testing data are taken on a grid of $5^6 = 15625$ equispaced points on $[0, 1)^6$ to evaluate the RMSEs. The numerical results are summarized as follows:

1. for the IUD data without any selection procedure, the RMSEs of the IK, EC, MC, and UC models are 1.67, 2.31, 2.26, and 2.49, respectively;
2. for the IUD data with the proposed selection procedure, the RMSEs of the EC, MC, and UC models are 1.52, 1.46, and 1.41, respectively;
3. for the OCSLHD data with the proposed selection procedure, the RMSEs of the EC, MC, and UC models are 1.39, 1.21, and 1.21, respectively.

It is clear that when the IUD is used as the design, the integral kriging models without any selection procedure perform poorly in terms of the RMSEs, even worse than the IK model; while with the proposed selection procedure, the integral kriging models significantly improve the prediction accuracies: not only better than the ones without any selection procedure, but also better than the IK model. Moreover, the OCSLHD associated with the proposed selection procedure yields more promising results than the IUD associated with the proposed selection procedure. This demonstrates the effectiveness of the interface between the OCSLHD and the proposed selection procedure.

This example assumes that there is only one qualitative variable with four levels. Our method can also apply to the cases where there are multiple qualitative variables. Following the suggestion raised by one referee, we rewrite the true responses $\hat{y}_{2,\exp}(1, \mathbf{x})$, $\hat{y}_{2,1}(2, \mathbf{x})$, $\hat{y}_{2,2}(3, \mathbf{x})$ and $\hat{y}_{2,3}(4, \mathbf{x})$ as $\hat{y}_{2,\exp}(1, 1, \mathbf{x})$, $\hat{y}_{2,1}(2, 1, \mathbf{x})$, $\hat{y}_{2,2}(1, 2, \mathbf{x})$ and $\hat{y}_{2,3}(2, 2, \mathbf{x})$, respectively. That is, we now treat this experiment as the one with two qualitative variables each of which has two levels, and the correlation function (4) is adopted for the integral kriging models. Similar to the previous scenario, the proposed selection procedure chooses only $\hat{y}_{2,3}(2, 2, \mathbf{x})$ as the AR for the integral kriging models no matter the design is the IUD or OCSLHD. The numerical results of this scenario are summarized as follows:

1. for the IUD data without any selection procedure, the RMSEs of the IK, EC, MC, and UC models are 1.67, 2.43, 2.26, and 2.28, respectively;
2. for the IUD data with the proposed selection procedure, the RMSEs of the EC, MC, and UC models are 1.48, 1.46, and 1.43, respectively;
3. for the OCSLHD data with the proposed selection procedure, the RMSEs of the EC, MC, and UC models are 1.34, 1.22, and 1.22, respectively.

The conclusion is similar to the previous scenario.

7. CONCLUDING REMARKS AND DISCUSSIONS

Computer experiments with both qualitative and quantitative variables (BQQV) have received much attention in the recent literature. Modeling methods for such experiments have attempted to use the information of *all* responses corresponding to different level-combinations of the qualitative variables. However, such a modeling strategy is valid, only if all the responses are *similar*. Discussions and examples in this work demonstrate that the information of the auxiliary responses (ARs) may reduce the prediction accuracy of the target response (TR) when they are not similar, and this issue is often ignored by the existing design and modeling framework. To select the ARs similar to the TR, this work proposes a new interface on design and analysis for computer experiments with BQQV. To measure the similarities among the responses, the optimal clustered-sliced Latin hypercube designs (OCSLHDs) are proposed. Such designs are one kind of sliced Latin hypercube design with points clustered in the design region, and possess good uniformity for each slice. Based on the similarity measures produced by the OCSLHDs, a selection procedure is developed to further determine which ARs are really useful for the TR. Then, these ARs are included into the model to predict the TR. Empirical observations show that the interface between the OCSLHDs and the selection procedure is effective in terms of prediction accuracy. As pointed

out by one referee, our new interface on design and analysis can be a tool for model diagnostics. That is, our method diagnoses whether the model is valid for all the responses of different level-combinations of the qualitative variables or just some subsets of them.

The simulated examples in Section 5 show that the new interface on design and analysis is effective whether the responses are similar or not. This is because the selection procedure, based on the similarity measures produced by the OCSLHDs, helps to retain those ARs similar to the TR and filter out those that are not similar to the TR. Therefore, if there are some ARs that are not similar to the TR, the proposed interface on design and analysis is more promising than directly building the metamodels with the commonly used designs, as in Qian, Wu, and Wu (2008), Han et al. (2009), and Zhou, Qian, and Zhou (2011). On the other hand, if the experimenter has the prior knowledge that any specific pair of the responses are similar, then those direct approaches with space-filling designs, such as sliced Latin hypercube designs or their variations (e.g., sliced orthogonal array based Latin hypercube designs proposed by Hwang, He, and Qian 2015), are expected to be more appropriate. In practice, however, such knowledge is typically not available. Thus, the proposed interface on design and analysis is recommended to prevent including irrelevant information for the TR.

The multi-fidelity experiment in Section 6 used the same number of design points for each accuracy of the computer code. A more general situation may have more design points for the low accuracy responses than those for the high accuracy responses (see Kennedy and O'Hagan 2000; Qian et al. 2006). This is because the lower the accuracy is, the faster it runs, thus one can collect more data from low accuracy responses. Under such a situation, the proposed interface on design and analysis is not applicable since it requires the same number of design points for each level of the qualitative variable. It is nevertheless necessary for this situation to select the computer responses that are informative about the physical process, and this important issue is also ignored by most existing methods. To our knowledge, the ANOVA kriging suggested by Han et al. (2010) in an unpublished article is the only method that concerns this issue. However, the ANOVA kriging still needs further improvements and other new relevant methods also deserve further study.

The computing issues are also important for computer experiments with BQQV. Kriging with BQQV typically involves many parameters, which could result in a near-singular correlation matrix $\mathbf{R}$. The near-singular matrix makes its inverse rather challenging. In our work, a modified Design and Analysis of Computer Experiments (DACE) MATLAB toolbox of Zhou, Qian, and Zhou (2011) is used to build up the GP models with BQQV. We make use of the "nugget" technique in the DACE, which adds a small positive constant to each element on the diagonal of $\mathbf{R}$ to avoid the singularity problem. Also note that when there are too many parameters to be estimated, estimator may be trapped in a local optimum. To alleviate this problem, we use multiple initial values for the EC model, and the parameters estimated by the EC model can then be used as the initial values for MC and UC models (Zhou, Qian, and Zhou 2011). In our work, computer experiments with a small number of observations and factors are dealt with. In such cases, the "nugget" technique is effective in computing the inverse of matrices. When there are a large number of observations and/or factors, the "nugget" technique may not work so well. One possible solution is to first use the multi-step interpolation technique for accurate metamodeling (Floater and Iske 1996; Haaland and Qian 2011), then incorporate the proposed interface on design and analysis into such a modeling strategy to further enhance the prediction accuracy for computer experiments with BQQV.

We realize that computer experiments with BQQV are yet in an immature research stage and there are many issues that need to be resolved. Besides the issues mentioned above, more potential research directions that we also think deserve further investigations are pointed out in Supplementary Section S11. We recommend the readers to peruse them and sincerely hope that they can be addressed in the near future.

ACKNOWLEDGMENTS

We are grateful to the editor, associate editor, and two reviewers for their insightful comments and constructive suggestions. This work was supported by the National Natural Science Foundation of China (Grant Nos. 11271205, 11431006, 11101024, and 11471172), the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20130031110002), and the "131" Talents Program of Tianjin. The authorship is listed in alphabetic order.

SUPPLEMENTARY MATERIALS

Additional details: proof of Proposition 1 and additional details mentioned in the main text (pdf file).

Computer code: Matlab codes for implementing the methodology proposed in this article. Data for all examples are provided (zip file).

[Received December 2012. Revised August 2015.]

REFERENCES

Chen, H., Huang, H. Z., Lin, D. K. J., and Liu, M. Q. (2014), "Uniform Sliced Latin Hypercube Designs," unpublished manuscript. [8]

Dewettinck, K., Visscher, A. D., Deroo, L., and Huyghebaert, A. (1999), "Modeling the Steady-State Thermodynamic Operation Point of Top-Spray Fluidized Bed Processing," *Journal of Food Engineering*, 39, 131–143. [10]

Fang, K. T., Li, R., and Sudjianto, A. (2006), *Design and Modeling for Computer Experiments*, New York: Chapman & Hall/CRC Press. [1,2,6,7]

Fang, K. T., Lin, D. K. J., Winker, P., and Zhang, Y. (2000), "Uniform Design: Theory and Application," *Technometrics*, 42, 237–248. [6]

Floater, M. S., and Iske, A. (1996), "Multistep Scattered Data Interpolation Using Compactly Supported Radial Basis Functions," *Journal of Computational and Applied Mathematics*, 73, 65–78. [12]

Haaland, B., and Qian, P. Z. G. (2011), "Accurate Emulators for Large-scale Computer Experiments," *The Annals of Statistics*, 39, 2974–3002. [12]

Han, G., Notz, W. I., Santner, T. J., and Long, J. P. (2010), "ANOVA Kriging: A Methodology for Predicting the Output From a Complex Computer Code Having Quantitative and Qualitative Inputs," unpublished manuscript. [1,7,9,10,12]

Han, G., Santner, T. J., Notz, W. I., and Bartel, D. L. (2009), "Prediction for Computer Experiments Having Quantitative and Qualitative Input Variables," *Technometrics*, 51, 278–288. [1,12]

Handcock, M. S. (1991), "On Cascading Latin Hypercube Designs and Additive Models for Experiments," *Communications in Statistics: Theory and Methods*, 20, 417–439. [6]

Hickernell, F. J. (1998), "A Generalized Discrepancy and Quadrature Error Bound," *Mathematics of Computation*, 67, 299–322. [5,6]

Hwang, Y., He, X., and Qian, P. Z. G. (2015), "Sliced Orthogonal Array Based Latin Hypercube Designs," *Technometrics*, available at *http://dx.doi.org/10.1080/00401706.2014.993092*. [12]

Johnson, M. E., Moore, L. M., and Ylvisaker, D. (1990), "Minimax and Maximin Distance Designs," *Journal of Statistical Planning and Inference*, 26, 131–148. [6]

Joseph, V. R., and Delaney, J. D. (2007), "Functionally Induced Priors for the Analysis of Experiments," *Technometrics*, 46, 1–11. [1,2]

Kennedy, M. C., and O'Hagan, A. (2000)," Predicting the Output From a Complex Computer Code When Fast Approximations are Available," *Biometrika*, 87, 1–13. [12]

Li, W. W., and Wu, C. F. J. (1997), "Columnwise–Pairwise Algorithms With Applications to the Construction of Supersaturated Designs," *Technometrics*, 39, 171–179. [6]

Long, J. P., and Bartel, D. L. (2006), "Surgical Variables Affect the Mechanics of a Hip Resurfacing System," *Clinical Orthopaedics and Related Research*, 453, 115–122. [1]

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979), "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code," *Technometrics*, 44, 230–241. [3,8]

McMillian, N. J., Sacks, J., Welch, W. J., and Gao, F. (1999), "Analysis of Protein Activity Data by Gaussian Stochastic Process Models," *Journal of Biopharmaceutical Statistics*, 9, 145–160. [1,2]

Morris, M. D., and Mitchell, T. J. (1995), "Exploratory Design for Computational Experiments," *Journal of Statistical Planning and Inference*, 43, 381–402. [6]

Qian, P. Z. G. (2012), "Sliced Latin Hypercube Designs," *Journal of the American Statistical Association*, 107, 393–399. [5,8]

Qian, P. Z. G., Seepersad, C. C., Joseph, V. R., Allen, J. K., and Wu, C. F. J. (2006), "Building Surrogate Models Based on Detailed and Approximate Simulations," *ASME Transactions, Journal of Mechanical Design*, 128, 668–677. [1,7,12]

Qian, P. Z. G., Tang, B., and Wu, C. F. J. (2009), "Nested Space-Filling Designs for Experiments with Two Levels of Accuracy," *Statistica Sinica*, 19, 287–300. [1]

Qian, P. Z. G., Wu, H., and Wu, C. F. J. (2008), "Gaussian Process Models for Computer Experiments With Qualitative and Quantitative Factors," *Technometrics*, 50, 383–396. [1,2,7,12]

Qian, P. Z. G., and Wu, C. F. J. (2008), "Bayesian Hierarchical Modeling for Integrating Low-Accuracy and High-Accuracy Experiments," *Technometrics*, 50, 192–204. [1]

Rawlinson, J. J., Furman, B. D., Li, S., Wright, T. M., and Bartel, D. L. (2006), "Retrieval, Experimental, and Computational Assessment of the Performance of Total Knee Replacements," *Journal of Orthopaedic Research*, 24, 1384–1394. [1]

Santner, T. J., Williams, B. J., and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer. [1,2,3,6]

Shewry, M. C., and Wynn, H. P. (1987), "Maximum Entropy Sampling," *Journal of Applied Statistics*, 14, 165–170. [6]

Stone, M. (1977), "Asymptotics for and Against Cross-Validation," *Biometrika*, 64, 29–35. [8]

Ye, K. Q., Li, W., and Sudjianto, A. (2000), "Algorithmic Construction of Optimal Symmetric Latin Hypercube Designs," *Journal of Statistical Planning and Inference*, 90, 145–159. [6]

Zhou, Q., Qian, P. Z. G., and Zhou, S. (2011), "A Simple Approach to Emulation for Computer Models With Qualitative and Quantitative Factors," *Technometrics*, 53, 266–273. [1,2,12]