Applied Stochastic Models in Business and Industry

Special Issue Paper

Received 11 October 2015, Revised 6 August 2016, Accepted 8 August 2016 Published online 4 September 2016 in Wiley Online Library

(wileyonlinelibrary.com) DOI: 10.1002/asmb.2192

Uniform sliced Latin hypercube designs

Hao Chen[a], Hengzhen Huang[b], Dennis K. J. Lin[c*†] and Min-Qian Liu[d]

Sliced Latin hypercube designs (SLHDs) achieve maximum stratification in each dimension, but neither the full designs nor their slices can guarantee a good uniformity over the experimental region. Although the uniformity of the full SLHD and that of its slices are related, there is no one-to-one correspondence between them. In this paper, we propose a new uniformity measure for SLHDs by combining the two kinds of uniformity. Based on such a combined uniformity measure, the obtained uniform SLHDs have the design points evenly spread over the experimental region not only for the whole designs but also for their slices. Numerical simulation shows the effectiveness of the proposed uniform SLHDs for computer experiments with both quantitative and qualitative factors.
Copyright © 2016 John Wiley & Sons, Ltd.

Keywords: computer experiments; centered L_2-discrepancy; threshold accepting algorithm; uniformity

1. Introduction

Computer experiments are widely used in modern business, engineering, and sciences. Latin hypercube designs (LHDs) proposed by McKay, Beckman, and Conover [1] are commonly used for computer experiments because of their appealing marginal property. An $m\times q$ matrix is called an LHD, denoted by $L(m, q)$, if each of its q columns includes m equally spaced levels, say $1, \ldots, m$. Such a design achieves the maximum stratification when projected onto any univariate dimension. Many improvements on LHDs have been investigated [2, 3], where all the factors are typically quantitative.

Recently, computer experiments with both quantitative and qualitative factors have raised increasing interests. To suit such a computer experiment, Qian [4] first proposed a general method for constructing sliced Latin hypercube designs (SLHDs). An $n\times q$ matrix S is called an SLHD with s slices, denoted by $SL(n, q, s)$, if S is an $L(n, q)$ and can be partitioned into s slices each of which is an $L(m, q)$ with $m = n/s$ after the n levels are collapsed to m equally spaced levels according to $\lceil i/s\rceil$ for level i, where $\lceil a\rceil$ means the smallest integer greater than or equal to a. SLHDs inherit the good property of LHDs, that is, they possess maximum stratification in any one dimension as well as their slices. Further studies on SLHDs include some constructions ensuring good projection in more than one dimension or orthogonality between columns, that is, Yang, Lin, Qian, and Lin [5]; Huang, Yang, and Liu [6]; Cao and Liu [7]; and Yang, Chen, Lin, and Liu [8] proposed methods to construct orthogonal and nearly orthogonal SLHDs. Yin, Lin, and Liu [9] constructed SLHDs with an attractive low-dimensional stratification via orthogonal arrays, and Yang, Chen, and Liu [10] obtained SLHDs based on resolvable orthogonal arrays.

There is another disadvantage of SLHDs that remains to be addressed: SLHDs usually do not possess a good uniformity over the experimental region. In fact, the design points of an SLHD can be distributed along the diagonal or off-diagonal line of the experimental region. The following $SL(12, 2, 2)$, denoted by A, is a good example:

$$A = \left(\begin{array}{cccccc|cccccc} 11 & 8 & 1 & 6 & 4 & 10 & 7 & 12 & 3 & 2 & 9 & 5 \\ 1 & 3 & 11 & 7 & 9 & 6 & 4 & 2 & 8 & 10 & 12 & 5 \end{array}\right)^T.$$

The scatter plot of A is presented in Figure 1, in which the symbols '*' and '+' denote the design points of the first slice and the second slice, respectively.

[a] *Department of Statistics, Tianjin University of Finance and Economics, Tianjin 300222, China*
[b] *Department of Biostatistics, Biomathematics and Bioinformatics, Georgetown University, Washington, DC 20057, U.S.A.*
[c] *Department of Statistics, The Pennsylvania State University, University Park, PA 16802, U.S.A.*
[d] *LPMC and Institute of Statistics, Nankai University, Tianjin 300071, China*
**Correspondence to: Dennis K. J. Lin, Department of Statistics, The Pennsylvania State University, University Park, PA 16802, U.S.A.*
[†] *E-mail: dkl5@psu.edu*

Copyright © 2016 John Wiley & Sons, Ltd.

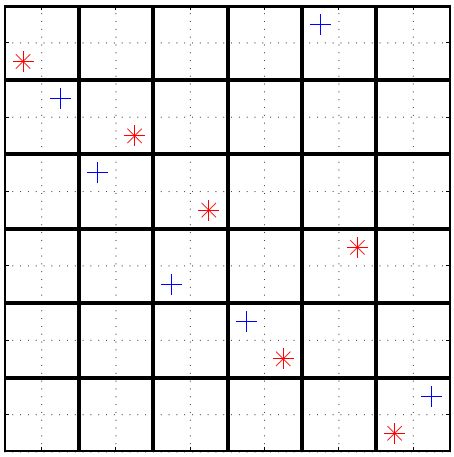

Figure 1. Scatter plot of A.

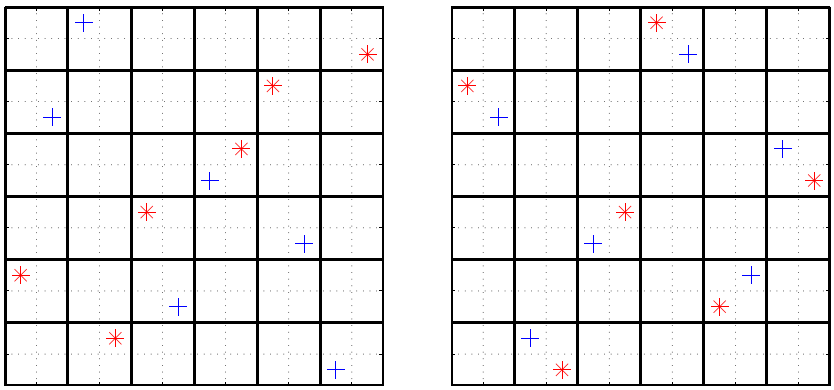

Figure 2. Scatter plots of B and C.

Obviously, design A achieves the maximum stratification when projected onto any one dimension, but its points are spread almost on the off-diagonal line in the two-dimensional region. Such a disadvantage goes against the space-filling principle and is undesirable for computer experiments. Although Yang, Chen, and Liu [10] considered such a problem, the existence of their designs depend heavily on the existence of the resolvable orthogonal arrays, whose numbers of runs and factors are constrained.

Note that in applications of computer experiments with both qualitative and quantitative factors, each slice of an SLHD corresponds to one level combination of the qualitative factors. Thus, the design points under each level combination of the qualitative factors should spread evenly over the experimental region when the response surfaces at different level combinations of the qualitative factors are similar [11]. However, there is no one-to-one correspondence between the uniformity of a whole SLHD and that of its slices. Two $SL(12, 2, 2)$'s, B and C, are displayed in Figure 2 as an example.

$$B = \left(\begin{array}{cccccc|cccccc} 9 & 1 & 5 & 8 & 12 & 4 & 2 & 3 & 10 & 11 & 7 & 6 \\ 10 & 4 & 6 & 8 & 11 & 2 & 9 & 12 & 5 & 1 & 7 & 3 \end{array}\right)^T \text{ and}$$

$$C = \left(\begin{array}{cccccc|cccccc} 1 & 7 & 12 & 4 & 9 & 6 & 5 & 10 & 2 & 8 & 3 & 11 \\ 10 & 12 & 7 & 1 & 3 & 6 & 5 & 4 & 9 & 11 & 2 & 8 \end{array}\right)^T.$$

It is clear that B has a good uniformity for the whole design, but poor for each slice; while C has a poor uniformity for the whole design, but good for each slice. In order to avoid the possible inconsistency between the uniformity of the whole design and that of its slices, it is desirable to develop a method to systematically generate a uniform SLHD in terms of both the whole design and its slices. To achieve such a goal, Ba, Brenneman, and Myers [12] obtained space-filling SLHDs by proposing a weighted average measure of the ϕ_r values of the whole design and its slices, where ϕ_r is the average reciprocal inter-point distance of a design. For more details, please refer to [12]. In this paper, we propose a new optimization criterion by combining the two uniformity measures of the whole design and its slices. The design obtained

Copyright © 2016 John Wiley & Sons, Ltd.

under such a combined measure, called a uniform sliced Latin hypercube design (USLHD), not only has good uniformity in terms of the whole design but also spreads the points of each slice evenly over the experimental region.

The remainder of this paper is organized as follows. Section 2 develops a new criterion that jointly considers the uniformity of the whole design and that of its slices, and then generates USLHDs based on such a new criterion. Section 3 conducts two simulated examples to illustrate the performance of USLHDs for computer experiments with both quantitative and qualitative factors. Concluding remarks and discussion are provided in Section 4. Some USLHDs with small sizes are tabulated in the Appendix.

2. Generation of uniform sliced Latin hypercube design

In this section, we first develop a combined uniformity measure in Section 2.1; in Section 2.2, algorithms for generating USLHDs are proposed; and finally, the proper weight in the combined measure is discussed in Section 2.3.

2.1. A combined uniformity measure

Among the commonly used space-filling criteria [13], the centered L_2-discrepancy $\left(CD_2\right)$ proposed by [14] is used here for evaluating the uniformity of SLHDs because of its appealing properties, such as invariance under reordering the runs, relabeling the factors, or reflecting the points about any plane passing through the center of the unit cube [3]. The CD_2 value of a design $D = \left(d_{ij}\right)$ with N runs and q factors in $[0,1]^q$, denoted by $CD_2(D)$, can be calculated by ([3])

$$CD_2(D) = \left[\left(\frac{13}{12}\right)^2 - \frac{2}{N}\sum_{k=1}^{N}\prod_{l=1}^{q}\left(1+\frac{1}{2}\mid d_{kl}-0.5\mid -\frac{1}{2}\mid d_{kl}-0.5\mid^2\right)\right.$$
$$\left. +\frac{1}{N^2}\sum_{k=1}^{N}\sum_{j=1}^{N}\prod_{i=1}^{q}\left(1+\frac{1}{2}\mid d_{ki}-0.5\mid +\frac{1}{2}\mid d_{ji}-0.5\mid -\frac{1}{2}\mid d_{ki}-d_{ji}\mid\right)\right]^{\frac{1}{2}}. \tag{1}$$

Note that before calculating the CD_2 value of a design $E = (e_{ij})$ with levels $1, \ldots, N$, one should scale the levels into $[0,1]$ by the following transformation

$$d_{ij} = \frac{e_{ij}-0.5}{N}. \tag{2}$$

For example, after scaling the levels into $[0,1]$ by (2), the CD_2 values of A, B, and C can be easily computed through (1), that is, $CD_2(A) = 0.0833$, $CD_2(B) = 0.0498$, and $CD_2(C) = 0.0560$. Comparing these three CD_2 values with the corresponding scatter plots illustrates a fact that a design with a smaller CD_2 value has a better uniformity. Thus, it is possible to find a USLHD by looking for an SLHD with the smallest CD_2 value.

For an SLHD $D = \left(D_{(1)}^T, \ldots, D_{(s)}^T\right)^T$, let $CD_2(D)$ and $CD_2\left(D_{(i)}\right)$ be the CD_2 values of D and $D_{(i)}$, respectively, where $D_{(1)}, \ldots, D_{(s)}$ are the s slices of D. A combined uniformity measure can be of the following form:

$$CD_2(D,\xi) = \xi CD_2(D) + (1-\xi)\sqrt[s]{\prod_{i=1}^{s} CD_2\left(D_{(i)}\right)}, \tag{3}$$

where $0 \leqslant \xi \leqslant 1$ is a real weighting parameter and $\sqrt[s]{\prod_{i=1}^{s} CD_2\left(D_{(i)}\right)}$ is the geometric mean of $CD_2\left(D_{(i)}\right)$'s. Note that $CD_2(D)$ and $CD_2\left(D_{(i)}\right)$'s may have different magnitudes, which may bring unfair comparison for the two kinds of uniformity and difficulty for determining the proper value of ξ. Instead of directly using the CD_2 values as in (3), we need a measure that not only has the same magnitude order for the whole design and its slices but also can reflect the uniformity of designs. For this purpose, we introduce the uniformity efficiency (U-efficiency for short) of D, which is defined as

$$E_U(D) = \frac{CD_2(U)}{CD_2(D)}, \tag{4}$$

where U is a uniform design with the same parameters as D under the CD_2. Many uniform designs can be found on website http://uic.edu.hk/isci/UniformDesign/UniformDasign.html. The uniform design tables in the website were constructed by Professor Fang, the co-founder of the uniform designs, and his collaborators via various combinatorial and computational methods, and they are universally recognized as the 'optimal' uniform designs with the smallest discrepancies. It is clear

Copyright © 2016 John Wiley & Sons, Ltd.

that $0 < E_U(D) \leqslant 1$ and $E_U(D)$ has the same order as $E_U\left(D_{(i)}\right)$'s, where $E_U(D)$ and $E_U\left(D_{(i)}\right)$'s are the U-efficiencies of D and $D_{(i)}$ for $i = 1, \ldots, s$, respectively. The proposed combined uniformity measure is then of the following form:

$$E_U(D, \omega) = \omega E_U(D) + (1 - \omega)\sqrt[s]{\prod_{i=1}^{s} E_U\left(D_{(i)}\right)}, \tag{5}$$

where $0 \leqslant \omega \leqslant 1$. Because $CD_2(U)$ is fixed, the larger the $E_U(D)$ is, the smaller the $CD_2(D)$ is. Thus, for a fixed ω, the objective is to find a design $D^* \in \mathcal{D}$ such that

$$E_U(D^*, \omega) = \max_{D \in \mathcal{D}} \left(E_U(D, \omega)\right). \tag{6}$$

Here, D^* is called a USLHD. Optimization algorithms for finding a D^* are given in the next subsection.

2.2. *Optimization algorithms*

As a stepping stone to USLHDs, the neighbor of an SLHD is an important concept that will be used in the proposed optimization procedure. Let $\mathcal{D}$ be the set consisting of all the $SL(n, q, s)$'s, then a neighbor of an SLHD $D_0 \in \mathcal{D}$ can be constructed by the following algorithm.

Algorithm 1

Step 1. Randomly choose one column of D_0, then from each slice of this column, choose one element such that these s elements, say $a_1, \ldots, a_s$, are 'equal' in the sense that $\lceil a_1/s \rceil = \cdots = \lceil a_s/s \rceil$.
Step 2. Randomly choose two elements among $a_1, \ldots, a_s$, say a_{i_1} and a_{i_2}, exchange their positions in the column.
Step 3. Randomly choose one of a_{i_1} and a_{i_2}, say a_{i_1}, and select an element from the same column in the same slice of a_{i_1}, say b_1 $\left(b_1 \neq a_{i_1}\right)$, exchange their positions.

The resulting design is one neighbor of D_0. It is obvious that such an exchanging procedure does not change the sliced structure of an SLHD. This is necessary for a design to be a neighbor of an SLHD.

To search USLHDs, we use the threshold accepting (TA) algorithm first described by Dueck and Scheuer [15]. The TA algorithm overcomes the problem of stopping in local minima for the classical local search by also accepting new solutions that lead to slightly worse objective function values. In addition, the TA algorithm is a modification of the simulated annealing algorithm [16] and has been shown to be simpler and more efficient than the simulated annealing algorithm in many applications [3]. The step-by-step guidelines for the proposed optimization algorithm are given as follows.

Algorithm 2

Step 1. Give $n,\ q,\ s,\ \omega$, randomly generate an $SL(n, q, s)$ as the initial design D_0 using the method in [4], and calculate $E_U(D_0, \omega)$. Set a non-positive sequence of threshold parameters $T_1 < \cdots < T_L = 0$. Denote the iteration number by I under each T_l for $l = 1, \ldots, L$. Set two indexes $l = 1,\ i = 1$.
Step 2. Randomly construct a neighbor of D_0 by Algorithm 1, denoted by D_c, and calculate $E_U(D_c, \omega)$.
Step 3. If $E_U(D_c, \omega) - E_U(D_0, \omega) \geqslant T_l$, replace D_0 by D_c; else leave D_0 unchanged.
Step 4. Update $i = i + 1$, if $i \leqslant I$, go to Step 2.
Step 5. Update $l = l + 1$, if $l \leqslant L$, reset $i = 1$ and go to Step 2; else output $D_{\text{best}} = D_0$.

Remark 1
Setting the sequence of $T_1, \ldots, T_L$ is a critical step. Several candidate sequences can be tried, and the one that can bring a quicker convergence and a more remarkable improvement is adopted.

The design obtained by Algorithm 2 is called a USLHD in terms of the combined uniformity with weight ω, denoted by $USL(n, q, s, \omega)$. It jointly considers the uniformity of the whole SLHD and that of its slices, so the design points of both the whole design and each slice are distributed evenly over the experimental region.

The resulting design D_{best} may be locally optimal depending on the selection of initial design. Hence, it is strongly recommended to run the algorithm a number of times with different initial designs and then select the best one among the resulting designs. Moreover, for an $SL(n, q, s)$, determining what a particular value ω_0 should be assigned to ω so that the $USL(n, q, s, \omega_0)$ is the most effective one among $SL(n, q, s, \omega)$'s is another important issue. This will be discussed next.

Copyright © 2016 John Wiley & Sons, Ltd.

2.3. Determination of the weight ω

From (5) and (6), as ω decreases from 1 to 0, $E_U(D^*)$ decreases, while $\sqrt[s]{\prod_{i=1}^{s} E_U\left(D^*_{(i)}\right)}$ increases, where $D^*_{(1)}, \ldots, D^*_{(s)}$ are the s slices of D^*. For such a trade-off, it is appropriate to avoid a low $E_U(D^*)$. So we impose a lower threshold l_u on $E_U(D^*)$, meanwhile maximizing $\sqrt[s]{\prod_{i=1}^{s} E_U\left(D^*_{(i)}\right)}$, and this leads to the following multi-objective optimization problem

$$\max_{\omega} \sqrt[s]{\prod_{i=1}^{s} E_U\left(D^*_{(i)}\right)}, \text{subject to } E_U(D^*) \geqslant l_u. \tag{7}$$

To solve (7), we introduce a tool called 'ω-trace', which plots $E_U(D^*)$ and $\sqrt[s]{\prod_{i=1}^{s} E_U\left(D^*_{(i)}\right)}$ as functions of ω. For convenience, we take a sequence of values $\{0, 0.05, 0.1, \ldots, 1\}$ for ω. As ω decreases from 1 to 0, $E_U(D^*)$ will decrease to the lower threshold l_u at some ω_0, and at the same time, $\sqrt[s]{\prod_{i=1}^{s} E_U\left(D^*_{(i)}\right)}$ takes the maximum value among all the ω's at which $E_U(D^*)$ is larger than l_u. That is, ω_0 is just the value of ω we are looking for. The following is an illustrative example.

Example 1

Suppose S_0 is a randomly generated $SL(18, 3, 3)$, then we carry out Algorithm 2 with 21 values of ω from 0 to 1 by 0.05 to search for the desired design, that is, $\omega_i = 0.05(i-1)$ for $i = 1, \ldots, 21$. In the algorithm, set the threshold parameters $T_1, \ldots, T_{11}$ to be $T_i = -10^{-5} + 10^{-6}(i-1)$ for $i = 1, \ldots, 11$. For each ω_i, we obtain a $USL\left(18, 3, 3, \omega_i\right)$, denoted by S_i for $i = 1, \ldots, 21$, and compute $E_U(S_i)$ and $\sqrt[3]{\prod_{j=1}^{3} E_U\left(S_{i(j)}\right)}$, where $S_{i(j)}$ for $j = 1, 2, 3$ are the three slices of S_i. To obtain the U-efficiencies, the CD_2 values of the corresponding uniform designs $U_{18}\left(18^3\right)$ and $U_6\left(6^3\right)$, which are 0.0506 and 0.1365, respectively, can be found from the website http://uic.edu.hk/isci/UniformDesign/UniformDasign.html, where $U_n\left(n^q\right)$ denotes a uniform design with n runs and q n-level factors. Then $E_U(S_i)$ and $\sqrt[3]{\prod_{j=1}^{3} E_U\left(S_{i(j)}\right)}$ for $i = 1, \ldots, 21$ and $j = 1, 2, 3$ can be computed through (4). Now, we plot the 'ω-trace', which is presented in Figure 3. As for the lower threshold of the U-efficiency for the whole design, we take the upper five percent quartile of the U-efficiencies of 10,000 randomly generated $SL(18, 3, 3)$'s, which is 0.7314, that is, $l_u = 0.7314$. Such a threshold can ensure that the obtained $USL(18, 3, 3, \omega)$ has a CD_2 smaller than about 95% randomly generated $SL(18, 3, 3)$'s when considering the uniformity of the whole design. From Figure 3, we find that as ω decreases to 0.45, the lower threshold has been reached; thus, we can assign $\omega = 0.45$ for this example.

Remark 2

Taking the upper five percent quartile of the U-efficiencies of a large number of randomly generated $SL(n, q, s)$'s as the lower threshold of the U-efficiency of the whole design is not the essence; some other values can also be taken according

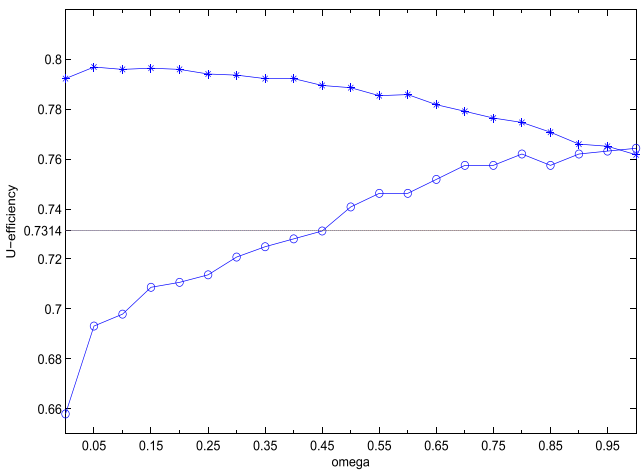

Figure 3. Plot of 'ω-trace', where the broken line with '$\circ$' corresponds to the U-efficiencies of the whole designs; and the broken line with '$*$' corresponds to the geometric means of the U-efficiencies of the slices.

Copyright © 2016 John Wiley & Sons, Ltd.

to the user's need, such as the upper two percent quartile, which can ensure that the obtained USLHD has a CD_2 smaller than about 98% of the randomly generated $SL(n, q, s)$'s when considering the uniformity of the whole design.

Remark 3
From Figure 3, we find that the lower threshold of the U-efficiency of the whole design is reached coincidentally at $\omega = \omega_{10} = 0.45$, which is just among the initial values taken for ω, that is, $\omega_1, \ldots, \omega_{21}$, in Example 1. In general, we suggest taking ω to be the smallest value among the initial ω_i's that are larger than the one at which the lower bound l_u is reached. To obtain a USLHD for any given size, we suggest 0.5 for ω if we cannot afford so much computational burden to determine the value of ω. In fact, the case of $\omega = 0.5$ gives approximately equal importance to both the uniformity of the whole design and that of its slices, which appears to be a fair choice.

3. Simulation

In this section, we provide two simulated examples to illustrate the performance of USLHDs when used for building Gaussian process (GP) models for computer experiments with both quantitative and qualitative factors. The computation is implemented by MATLAB toolbox DACE [17]. Before the simulation is conducted, we briefly introduce the GP model with both quantitative and qualitative factors. For more details, please refer to [18].

Assume that there are I quantitative factors $x_1, \ldots, x_I$ and J qualitative factors $z_1, \ldots, z_J$. Let $\mathbf{x} = (x_1, \ldots, x_I)$, $\mathbf{z} = (z_1, \ldots, z_J)$, then $\mathbf{w} = (\mathbf{x}, \mathbf{z})$ is the input vector. The GP model assumes the true function $y(\mathbf{w})$ is a realization from a stochastic process

$$y(\mathbf{w}) = \mathbf{f}(\mathbf{w})'\boldsymbol{\beta} + \varepsilon(\mathbf{w}), \tag{8}$$

where $\mathbf{f}(\mathbf{w}) = \left(f_1(\mathbf{w}), \ldots, f_p(\mathbf{w})\right)'$ is a vector of pre-specified regression functions, $\boldsymbol{\beta} = \left(\beta_1, \ldots, \beta_p\right)'$ is a vector of unknown coefficients, and the residual $\varepsilon(\mathbf{w})$ is assumed to be a stationary GP with mean 0 and variance σ^2. Because $\mathbf{f}(\mathbf{w})$ is rarely known before the experiment, a constant μ is frequently used instead of $\mathbf{f}(\mathbf{w})'\boldsymbol{\beta}$. Using a constant for the regression part of the GP model is referred to as *ordinary kriging* (OK). Suppose z_j has m_j levels, $j = 1, \ldots, J$. Then there are $s = \prod_{j=1}^{J} m_j$ level combinations for the qualitative factors, say $c_1, \ldots, c_s$. For two input values $\mathbf{w}_1 = \left(\mathbf{x}_1, \mathbf{z}_1\right) = \left(\mathbf{x}_1, c_u\right)$ and $\mathbf{w}_2 = \left(\mathbf{x}_2, \mathbf{z}_2\right) = \left(\mathbf{x}_2, c_v\right)$, $1 \leqslant u, v \leqslant s$, the correlation function between $\varepsilon\left(\mathbf{w}_1\right)$ and $\varepsilon\left(\mathbf{w}_2\right)$ can be defined as

$$R(\mathbf{w}_1, \mathbf{w}_2) = \operatorname{cor}(\varepsilon(\mathbf{w}_1), \varepsilon(\mathbf{w}_2)) = \tau_{c_u, c_v} \exp\left\{ -\sum_{i=1}^{I} \theta_i |x_{1i} - x_{2i}|^2 \right\}, \tag{9}$$

where $\tau_{c_u, c_v} \in (-1, 1)$ is the cross correlation between responses corresponding to level combinations c_u and c_v, and $\theta_i \in (0, +\infty)$ is called a scale parameter. All these parameters are estimated by maximum likelihood. Qian, Wu, and Wu [18] argued that the correlation function (9) is valid under the condition that the $s \times s$ matrix $\mathbf{T} = \left(\tau_{c_u, c_v}\right)$ is a positive definite matrix with unit diagonal elements, $1 \leqslant u, v \leqslant s$. Several choices of the τ_{c_u, c_v} in the literature satisfy this condition [19–21]. Empirical evidence suggests using the method of [21] for modeling the cross correlation [11, 22]. Here, we will adopt this method as well.

In a computer experiment, the output is observed at N input values given by the rows of $\mathbf{W} = \left(\mathbf{w}_1, \ldots, \mathbf{w}_N\right)^T$. This yields an $N \times 1$ vector $\mathbf{Y}$ of observed outputs. Based on the data, the best linear unbiased predictor of the OK model at an untried site $\mathbf{w}^*$ can be constructed via ([2])

$$\hat{y}\left(\mathbf{w}^*\right) = \hat{\mu} + \mathbf{r}^T\mathbf{R}^{-1}\left(\mathbf{Y} - \mathbf{1}_N\hat{\mu}\right), \tag{10}$$

where $\mathbf{r} = \left(R\left(\mathbf{w}^*, \mathbf{w}_1\right), \ldots, R\left(\mathbf{w}^*, \mathbf{w}_N\right)\right)^T$, $\mathbf{R} = \left(R\left(\mathbf{w}_i, \mathbf{w}_j\right)\right)_{N \times N}$, $\mathbf{1}_N$ is an $N \times 1$ vector of ones, and $\hat{\mu} = \left(\mathbf{1}_N^T\mathbf{R}^{-1}\mathbf{1}_N\right)^{-1}$ $\mathbf{1}_N^T\mathbf{R}^{-1}\mathbf{Y}$. To assess the accuracy of the best linear unbiased predictor in (10), the root mean squared prediction error (RMSPE) is used. For N_0 testing points $\mathbf{w}_1^*, \ldots, \mathbf{w}_{N_0}^*$, the RMSPE is defined as

$$\text{RMSPE} = \sqrt{\frac{1}{N_0}\sum_{u=1}^{N_0}\left(y\left(\mathbf{w}_u^*\right) - \hat{y}\left(\mathbf{w}_u^*\right)\right)^2}. \tag{11}$$

Example 2
In this example, we explore how the weight ω in (5) affects the prediction performance of the USLHDs with different parameters generated by Algorithm 2. Suppose the true response surfaces for different levels of the qualitative factor are similar to each other as follows:

Copyright © 2016 John Wiley & Sons, Ltd.

$$f_1(x_1,x_2,x_3,z)=\begin{cases}\exp\left(1+2x_1\right)+\sin\left(3\pi\left(x_2+x_3\right)\right), & \text{if } z=1,\\ \exp\left(1+2x_1\right)+\sin\left(5\pi\left(x_2+x_3\right)\right), & \text{if } z=2,\\ \exp\left(1+3x_1\right)-\cos\left(7\pi\left(x_2+x_3\right)\right), & \text{if } z=3.\end{cases}$$

To generate USLHDs, the threshold parameters in Algorithm 2 are set to range from -10^{-5} to 0 by 10^{-6}, and the iteration number I is set to be 10,000. Before carrying out the experiment, the levels of x_1, x_2, x_3 should be scaled into $[0,1]^3$ by (2). For building the GP model, we use the OK model and the cross correlation method in [21] for (9). For validation, we repeat 100 times for generating USLHDs from random initial SLHDs; building GP models and predicting at 300 Latin hypercube sample points that can be generated in $[0,1]^3$ for x_1, x_2, and x_3 by a function *lhsdesign* in MATLAB. Such 300 points are randomly divided into three groups, each of which having 100 points corresponds to one of the three levels 1, 2, 3 of z. The mean and standard deviation (SD) values of RMSPEs are provided in Table I, as well as the averaged CD_2 values of the whole designs and their slices, denoted by $CD_2(D)$, and $CD_2\left(D_{(i)}\right)$ for $i=1,2,3$, respectively.

From Table I, we can see that USLHDs with $\omega=0$, that is, only focusing on the uniformity of the slices, have the smallest mean and SD values of RMSPEs. In addition, the mean and SD values of RMSPEs show an increasing trend as the weight ranging from 0 to 1.

Next, how about the performance of the USLHDs when the true response surfaces have significant differences? Suppose the true response surfaces for different levels of the qualitative factor are given as follows:

$$f_2\left(x_1,x_2,x_3,z\right)=\begin{cases}\sin\left(x_1+x_2+x_3\right), & \text{if } z=1,\\ \exp\left(x_1+x_2-x_3\right), & \text{if } z=2,\\ \left(1+3x_1\right)^2+x_2^4-x_3^3, & \text{if } z=3.\end{cases}$$

Table II shows a different trend of RMSEs as ω ranging from 0 to 1, and obviously, USLHDs with $\omega=1$, that is, only focusing on the uniformity of the whole design, perform the best this time.

From Tables I and II, it is observed that for similar response functions corresponding to different slices, a smaller ω value brings a smaller mean of RMSPEs, that is, it is more important that the individual slices be space filling, but less important that the overall design be space filling; while for dissimilar response functions corresponding to different slices, a larger ω value brings a smaller mean of RMSPEs, that is, it is more important that the overall design be space filling and less important that the individual slices be space filling. Note that this is only a conjecture/observation based on the two cases of f_1 and f_2; additional study is needed to confirm this.

It should be pointed out that the real response functions are usually unknown in practice, so we suggest again to use USLHDs with $\omega=0.5$ for the sake of robustness. In the next example, we will compare the prediction performance of USLHDs with $\omega=0.5$, optimal SLHDs (OptSLHDs, [12]) and ordinary SLHDs [4]. Denote OptSLHDs with n rows, q columns, and s slices by $OptSL(n,q,s)$.

Remark 4

The boxplots of the estimated cross correlations among the responses under $z=1,2,3$ for f_1 and f_2 are presented in Figure 4(a) and (b), respectively. They will help to visualize how similar the curves are for different values of z.

Table I. RMSPEs and CD_2 values for the uniform sliced Latin hypercube designs with different parameters under f_1.

	RMSPEs		CD_2 values			
	Mean	SD	$CD_2(D)$	$CD_2\left(D_{(1)}\right)$	$CD_2\left(D_{(2)}\right)$	$CD_2\left(D_{(3)}\right)$
$USL(18,3,3,0)$	1.6287	0.2896	0.0644	0.1567	0.1566	0.1579
$USL(18,3,3,0.45)$	1.7879	0.4909	0.0546	0.1652	0.1647	0.1648
$USL(18,3,3,1)$	2.0513	0.6851	0.0535	0.1785	0.1787	0.1790
$USL(24,3,3,0)$	1.3885	0.2823	0.0522	0.1206	0.1215	0.1216
$USL(24,3,3,0.5)$	1.5944	0.4370	0.0428	0.1307	0.1303	0.1307
$USL(24,3,3,1)$	1.7833	0.4940	0.0421	0.1436	0.1439	0.1426
$USL(27,3,3,0)$	1.3268	0.2353	0.0474	0.1088	0.1090	0.1096
$USL(27,3,3,0.5)$	1.5381	0.3707	0.0391	0.1180	0.1178	0.1173
$USL(27,3,3,1)$	1.7543	0.5233	0.0384	0.1320	0.1316	0.1321

RMSPEs, root mean squared prediction errors; SD, standard deviation.

Copyright © 2016 John Wiley & Sons, Ltd.

Table II. RMSPEs and CD_2 values for the uniform sliced Latin hypercube designs with different parameters under f_2.

	RMSPEs		CD_2 values			
	Mean	SD	$CD_2(D)$	$CD_2\left(D_{(1)}\right)$	$CD_2\left(D_{(2)}\right)$	$CD_2\left(D_{(3)}\right)$
$USL(18,3,3,0)$	1.5511	1.3046	0.0644	0.1590	0.1558	0.1550
$USL(18,3,3,0.45)$	1.0854	1.1367	0.0542	0.1634	0.1643	0.1628
$USL(18,3,3,1)$	0.9420	0.9665	0.0528	0.1789	0.1814	0.1778
$USL(24,3,3,0)$	0.9520	1.1629	0.0520	0.1202	0.1210	0.1218
$USL(24,3,3,0.5)$	0.5651	0.7514	0.0425	0.1309	0.1304	0.1301
$USL(24,3,3,1)$	0.4103	0.2905	0.0419	0.1446	0.1442	0.1448
$USL(27,3,3,0)$	0.5290	0.8000	0.0473	0.1087	0.1092	0.1087
$USL(27,3,3,0.5)$	0.4147	0.5385	0.0387	0.1172	0.1173	0.1176
$USL(27,3,3,1)$	0.3050	0.1033	0.0381	0.1295	0.1303	0.1302

RMSPEs, root mean squared prediction errors; SD, standard deviation.

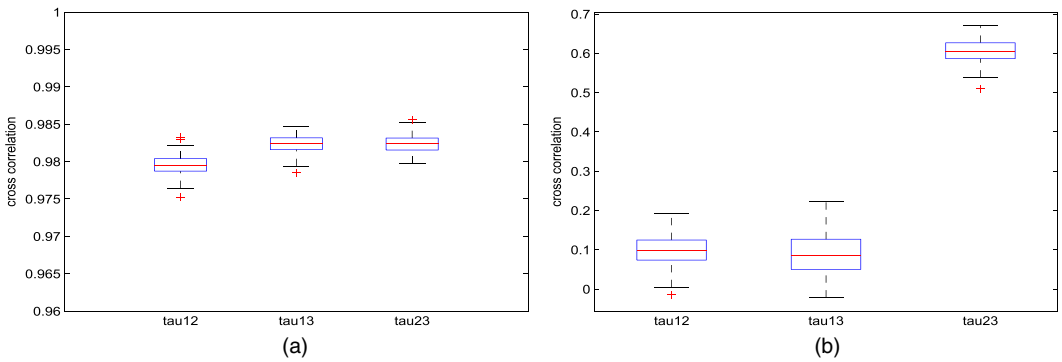

Figure 4. (a) Boxplots of the estimated cross correlations among the responses under $z = 1, 2, 3$ for f_1; (b) boxplots of the estimated cross correlations among the responses under $z = 1, 2, 3$ for f_2.

Table III. RMSPEs and CD_2 values for uniform SLHDs, optimal SLHDs, and SLHDs under f_3.

	RMSPEs		CD_2 values			
	Mean	SD	$CD_2(D)$	$CD_2\left(D_{(1)}\right)$	$CD_2\left(D_{(2)}\right)$	$CD_2\left(D_{(3)}\right)$
$USL(18,3,3,0.5)$	1.8109	0.8853	0.0541	0.1655	0.1669	0.1664
$OptSL(18,3,3)$	1.8689	1.0055	0.0603	0.1634	0.1634	0.1633
$SL(18,3,3)$	1.9570	0.8261	0.0802	0.1812	0.1820	0.1811
$USL(24,3,3,0.5)$	1.5071	0.5834	0.0427	0.1299	0.1296	0.1299
$OptSL(24,3,3)$	1.6290	0.7999	0.0489	0.1278	0.1283	0.1283
$SL(24,3,3)$	1.7051	0.5140	0.0684	0.1480	0.1469	0.1458
$USL(27,3,3,0.5)$	1.4683	0.2482	0.0390	0.1177	0.1178	0.1179
$OptSL(27,3,3)$	1.4616	0.5711	0.0448	0.1155	0.1160	0.1159
$SL(27,3,3)$	1.6368	0.4197	0.0643	0.1348	0.1344	0.1335

RMSPEs, root mean squared prediction errors; SD, standard deviation; SLHDs, sliced Latin hypercube designs.

Example 3

Suppose the true response surfaces for different levels of the qualitative factor are

$$f_3(x_1,x_2,x_3,z)=\begin{cases}\exp\left(1.5x_1\right)+\cos\left(3\pi\left(x_2+x_3\right)\right), & \text{if } z=1,\\ \left(1+2.5x_1\right)-\sin\left(5\pi\left(x_2+x_3\right)\right), & \text{if } z=2,\\ \exp\left(1+2x_1\right)+\cos\left(7\pi x_2\right)-\sin\left(7\pi x_3\right), & \text{if } z=3.\end{cases}$$

Copyright © 2016 John Wiley & Sons, Ltd.

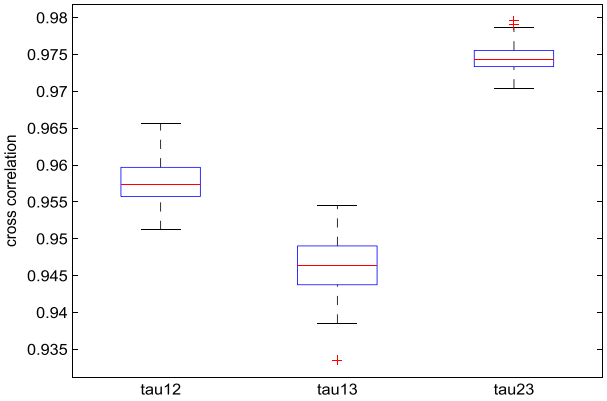

Figure 5. Boxplots of the estimated cross correlations among the responses under $z = 1, 2, 3$ for f_3.

The methods for building GP models and predicting are just the same as those in Example 2. The mean and SD values of RMSPEs are provided in Table III, as well as the averaged CD_2 values of the whole designs and their slices, denoted by $CD_2(D)$, and $CD_2\left(D_{(i)}\right)$ for $i = 1, 2, 3$, respectively.

It can be seen from Table III that both USLHDs and OptSLHDs outperform the random SLHDs, and USLHDs are comparable with OptSLHDs except that OptSLHDs have a little larger SD values.

Remark 5

The boxplot of the estimated cross correlations among the responses under $z = 1, 2, 3$ for f_3 is presented in Figure 5.

4. Concluding remarks and discussion

In this paper, we proposed a combined uniformity measure by combining the uniformity of the whole SLHD and that of its slices together, with a weighting parameter. This is a compromise between the two kinds of uniformity. Based on such a criterion, we constructed USLHDs by Algorithm 2. The obtained designs not only have good uniformity for the whole designs but also spread the design points of the slices evenly over the experimental region. To determine the proper value for the weighting parameter ω in the combined uniformity measure, a concept U-efficiency and a tool called 'ω-trace' were provided.

Yang, Lin, Qian, and Lin [5]; Huang, Yang, and Liu [6]; and Cao and Liu [7] mainly constructed orthogonal and nearly orthogonal SLHDs, but orthogonality does not guarantee a good uniformity. Yang, Chen, Lin, and Liu [8] generated orthogonal SLHDs and improved their overall space-filling property under the maximin distance criterion, but the uniformity of the slices was not considered. Yin, Lin, and Liu [9] only focused on low-dimensional projective stratification, and Yang, Chen, and Liu [10] only considered the uniformity of the whole SLHDs. Ba, Brenneman, and Myers [12] obtained space-filling SLHDs, which share some similarities with the proposed USLHDs. For example, not only the uniformity of the whole design but also the uniformity of their slices are considered. However, the uniformity measures are not the same, that is, ϕ_r extended from maximin distance measure in [12] and U-efficiency based on CD_2 in this paper, respectively. In addition, we also presented the 'ω-trace' method to determine the proper weight value for a USLHD, while they determined the weight subjectively.

The simulations indicated that USLHDs with better uniformity in terms of whole design tend to have better performance when the real response functions have big differences from each other, and USLHDs with better uniformity in terms of slices probably result in smaller RMSPEs when the real response functions are similar to each other. Thus, we suggest adopting $\omega = 0.5$ for the sake of robust application because the real response functions are usually unknown. The results in Example 3 showed that $USL(n, q, s, 0.5)$'s are comparable with OptSLHDs [12], and both of them outperform the random SLHDs [4]. The implementation of the proposed method and all examples in this paper are conducted by MATLAB codes, which are available from the authors upon request.

Appendix: Design tables of some uniform sliced Latin hypercube designs

Given subsequently are some USLHDs with small sizes because the limited space and long computing time. For more designs, we can provide the MATLAB codes.

Copyright © 2016 John Wiley & Sons, Ltd.

$$USL(12,2,2,0.5)=\left(\begin{array}{cccccc|cccccc}3&12&1&10&5&8&7&9&6&2&11&4\\6&4&9&7&1&12&5&2&8&3&10&11\end{array}\right)^T,$$

$$USL(12,3,2,0.5)=\left(\begin{array}{cccccc|cccccc}2&10&4&6&7&11&3&5&9&1&8&12\\5&11&7&1&10&3&2&12&9&8&4&6\\2&10&12&8&3&5&4&6&1&9&11&7\end{array}\right)^T,$$

$$USL(18,2,3,0.5)=\left(\begin{array}{cccccc|cccccc|cccccc}7&2&11&14&18&5&8&12&6&1&13&17&9&3&10&15&16&4\\10&15&18&2&9&5&1&6&17&8&11&14&13&3&7&16&4&12\end{array}\right)^T,$$

$$USL(18,3,3,0.5)=\left(\begin{array}{cccccc|cccccc|cccccc}5&9&17&3&12&14&6&13&11&8&18&2&10&1&7&16&15&4\\8&1&6&17&14&12&13&2&18&7&10&5&9&11&15&16&4&3\\18&12&3&4&9&15&2&5&17&8&11&14&1&10&13&7&16&6\end{array}\right)^T,$$

$$USL(24,2,3,0.5)=\left(\begin{array}{cccccccc|cccccccc|cccccccc}11&17&3&14&22&6&9&21&1&19&13&24&12&18&7&4&20&8&10&16&2&23&5&15\\15&7&3&4&22&16&21&10&11&2&18&14&9&20&6&23&17&13&1&12&19&5&8&24\end{array}\right)^T,$$

$$USL(24,3,3,0.5)=\left(\begin{array}{cccccccc|cccccccc|cccccccc}17&10&22&6&2&21&7&13&24&5&18&19&9&1&14&12&23&4&3&20&8&11&15&16\\8&17&3&19&11&23&6&14&16&2&5&12&24&15&20&9&10&7&22&18&13&4&1&21\\5&1&18&15&23&10&8&20&13&3&22&7&21&11&4&17&2&19&6&24&14&9&12&16\end{array}\right)^T,$$

$$USL(27,2,3,0.5)=\left(\begin{array}{ccccccccc|ccccccccc|ccccccccc}24&20&10&14&27&2&5&8&16&3&19&9&4&23&26&18&11&15&25&12&22&6&13&7&21&1&17\\24&5&2&20&11&25&8&15&17&4&14&23&18&7&19&26&12&1&3&27&16&21&10&6&22&13&9\end{array}\right)^T,$$

$$USL(27,3,3,0.5)=\left(\begin{array}{ccccccccc|ccccccccc|ccccccccc}3&19&12&6&26&9&13&23&17&14&4&22&25&2&21&7&11&18&10&8&27&1&15&24&5&20&16\\25&1&18&8&10&4&13&19&24&2&20&14&26&7&6&22&11&17&27&12&15&16&23&5&3&21&9\\7&10&19&24&18&14&5&2&25&23&21&26&15&16&6&9&1&11&17&27&8&13&4&20&3&22&12\end{array}\right)^T.$$

Acknowledgements

The authors thank the editor, an associated editor, and two referees for their valuable comments and suggestions. This work was supported by the National Security Agent (Grant No. H98230-15-1-0253), the National Natural Science Foundation of China (Grant Nos. 11271205, 11431006, 11471239, 11601367 and 11526147), 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.

References

1. McKay MD, Beckman RJ, Conover WJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 1979; **21**:239–245.
2. Santner TJ, Williams BJ, Notz WI. *The Design and Analysis of Computer Experiments*. Springer: New York, 2003.
3. Fang KT, Li R, Sudjianto A. *Design and Modeling for Computer Experiments*. CRC Press: New York, 2006.
4. Qian PZG. Sliced Latin hypercube designs. *Journal of the American Statistical Association* 2012; **107**:393–399.
5. Yang JF, Lin CD, Qian PZG, Lin DKJ. Construction of sliced orthogonal Latin hypercube designs. *Statistica Sinica* 2013; **23**:1117–1130.
6. Huang HZ, Yang JF, Liu MQ. Construction of sliced (nearly) orthogonal Latin hypercube designs. *Journal of Complexity* 2014; **30**:355–365.
7. Cao RY, Liu MQ. Construction of second-order orthogonal sliced Latin hypercube designs. *Journal of Complexity* 2015; **31**:762–772.
8. Yang JY, Chen H, Lin DKJ, Liu MQ. Construction of sliced maximin-orthogonal Latin hypercube designs. *Statistica Sinica* 2016; **26**:589–603.
9. Yin YH, Lin DKJ, Liu MQ. Sliced Latin hypercube designs via orthogonal arrays. *Journal of Statistical Planning and Inference* 2014; **149**: 162–171.
10. Yang X, Chen H, Liu MQ. Resolvable orthogonal array-based uniform sliced Latin hypercube designs. *Statistics and Probability Letters* 2014; **93**:108–115.
11. Huang HZ, Yang JF. Computer experiments with both qualitative and quantitative variables. *Technometrics* 2016. DOI: 10.1080/00401706.2015.1094416.
12. Ba S, Brenneman WA, Myers WR. Optimal sliced Latin hypercube designs. *Technometrics* 2015; **57**:479–487.
13. Fang KT, Lin DKJ, Winker P, Zhang Y. Uniform design: theory and application. *Technometrics* 2000; **42**:237–248.
14. Hickernell FJ. A generalized discrepancy and quadrature error bound. *Mathematics of Computation* 1998; **67**:299–322.
15. Dueck G, Scheuer T. Threshold accepting: a general purpose optimization algorithm superior to simulated annealing. *Journal of Computational Physics* 1990; **90**:161–175.
16. Kirkpatrick S, Gelett C, Vecchi M. Optimization by simulated annealing. *Science* 1983; **220**:621–630.

Copyright © 2016 John Wiley & Sons, Ltd.

17. Lophaven SN, Nielsen HB, Sondergaard J, DACE. *MATLAB Kriging Toolbox*. Version 2.5, available at http://www2.imm.dtu.dk/projects/dace/ [4 September 2002].
18. Qian PZG, Wu HQ, Wu CFJ. Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics* 2008; **50**:383–396.
19. Joseph VR, Delaney JD. Functionally induced prior for the analysis of experiments. *Technometrics* 2007; **49**:1–11.
20. McMillian NJ, Sacks J, Welch WJ, Cao F. Analysis of protein activity data by Gaussian stochastic process models. *Journal of Biopharmaceutical Statistics* 1999; **9**:145–160.
21. Rebonato R, Jackel P. The most general methodology for creating a valid correlation matrix for risk management and option pricing purposes. *The Journal of Risks* 1999; **2**:17–27.
22. Zhou Q, Qian PZG, Zhou SY. A simple approach to emulation for computer models with qualitative and quantitative factors. *Technometrics* 2011; **53**:266–273.

Copyright © 2016 John Wiley & Sons, Ltd.