

## 灵活的多层嵌套拉丁超立方体设计构造\*

徐 璘<sup>1</sup>, 段晓君<sup>1</sup>, 王正明<sup>2</sup>, 晏 良<sup>1</sup>

(1. 国防科技大学 文理学院, 湖南 长沙 410073; 2. 国防科技大学 前沿交叉学科学院, 湖南 长沙 410073)

**摘要:** 计算机试验的设计方法越来越受到重视, 嵌套拉丁超立方体设计是计算机试验设计中的一种新型方法, 其在多种精度试验中有广泛的应用。但多数嵌套拉丁超立方体设计要求低精度试验次数需为高精度试验次数的倍数, 这在应用中会有很大的局限性。通过对其构造方法的改进, 得到一种结构更加灵活的多层嵌套拉丁超立方体设计, 使得不同精度试验的次数可以更加灵活选取。该设计方法在一维投影上可以达到很好的均匀性。仿真结果表明, 该方法较若干相关方法能够达到更小的均方误差。

**关键词:** 嵌套设计; 计算机试验; 拉丁超立方体设计; 多层设计

**中图分类号:** O21 **文献标志码:** A **文章编号:** 1001-2486(2019)03-174-05

## A new construction for flexible multi-layer nested Latin hypercube designs

XU Jin<sup>1</sup>, DUAN Xiaojun<sup>1</sup>, WANG Zhengming<sup>2</sup>, YAN Liang<sup>1</sup>

(1. College of Liberal Arts and Sciences, National University of Defense Technology, Changsha 410073, China;

2. College of Advanced Interdisciplinary Studies, National University of Defense Technology, Changsha 410073, China)

**Abstract:** The designs for computer experiments attract more and more attention, and the nested Latin hypercube designs play significant roles in computer experiments, for they mainly deal with the experiments which have high-accuracy and low-accuracy parts. However, most nested Latin hypercube designs have the restriction that the run sizes of low-accuracy experiments should be the multiple of high-accuracy experiments. A general multi-layer nested Latin hypercube design which has flexible run sizes for different accuracy experiments was proposed by improving the traditional construction method. This design method can achieve the desirable one-dimensional uniformity. Numerical simulations show that this method perform well on the sampling variance.

**Keywords:** nested designs; computer experiment; Latin hypercube design; multi-layer design

随着计算机的高速发展, 有越来越多的试验在计算机上运行<sup>[1-3]</sup>。在计算机上进行试验存在诸多优势: 首先, 有些费时费力的试验可以在计算机上进行预先试验, 对其进行初步的研究和参数筛选; 其次, 对于一些现实中难以实现甚至无法实现的试验可以在计算机上进行。对某计算机模型, 令  $\mathbf{X}$  表示  $[0, 1]^p$  上的输入、 $f(\mathbf{X})$  表示该模型的输出, 则该模型的均值为  $\mu = \int_{[0, 1]^p} f(\mathbf{X}) d\mathbf{X}$ 。对于某  $n$  行设计矩阵  $\mathbf{D} \subset [0, 1]^p$ ,  $\hat{\mu} = (1/n) \sum_{x \in \mathbf{D}} f(x)$  是  $\mu$  的估计。显然, 设计矩阵的好坏对估计精度的高低有着至关重要的作用。对于  $n \times p$  的设计  $\mathbf{D}$ , 如果其每列均满足在任一区间  $[0, 1/n), \dots, [(n-1)/n, 1)$  中有且仅有一个设计点, 则称  $\mathbf{D}$  为拉丁超立方体设计 (Latin

Hypercube Designs, LHD)。因此, 由 LHD 得到的估计  $\hat{\mu}$  比随机抽样得到的要更精确<sup>[4]</sup>。文献[5]首先给出了一种拉丁超立方体设计的构造方法。

几乎所有的计算机试验都存在多种精度的情况, 如何将这多种试验结合起来并充分利用所有数据信息是十分关键的问题。多种精度的试验有许多的应用, 如飞行器设计相关试验等<sup>[6]</sup>。基于多种精度试验的建模方法也层出不穷, 如贝叶斯修正方法<sup>[7]</sup>、贝叶斯分层建模<sup>[8]</sup>和计算机试验与实体试验的联合建模<sup>[9]</sup>等。在多种精度试验的设计选点部分, 文献[10]提出了一种方法: 嵌套拉丁方设计 (Nested Latin Hypercube Designs, NLHD)。该设计保证了所有高精度的试验点在低精度试验中进行了同样的试验, 这对后续试验

\* 收稿日期: 2018-03-20

基金项目: 国家自然科学基金资助项目(61573367, 11771450)

作者简介: 徐璘(1990—), 男, 河南焦作人, 博士研究生, E-mail: xujin\_nudt@163.com;

段晓君(通信作者), 女, 教授, 博士, 博士生导师, E-mail: xj\_duan@163.com

的建模有很好的帮助<sup>[11]</sup>,同时不同精度下的试验点集均为 LHD。然而,SLHD 及其若干后续研究(例如文献[12]和文献[13])均限制低精度试验的试验次数必须是高精度的倍数。这就导致在具体的试验中,试验次数会有诸多限制。文献[14]和文献[15]介绍了结构灵活的设计,分别为序贯设计(Sequential Designs, SD)和广义嵌套拉丁超立方体设计(General Nested Latin Hypercube designs, GNLH),但 SD 方法的均匀性不够,其较低精度的试验点在大多数情况下不是 LHD,而 GNLH 只能构造两层的嵌套设计。因此,对于三种或三种以上精度的计算机试验,现有工作无法给出结构灵活的嵌套拉丁超立方体设计。

本文介绍一种结构灵活的多层嵌套拉丁超立方体设计(Multi-layer Nested Latin hypercube Designs, MND),其试验设计点数可以灵活选取,同时每类精度试验的试验点均有很好的二维投影均匀性。

## 1 MND 的结构与性质

### 1.1 MND 的构造方法

下面对 MND 构造算法中使用到的符号进行说明。 $Z_n$  表示整数集合  $\{1, \dots, n\}$ 。对于向量  $\mathbf{v}$ ,  $\mathbf{v}(i)$  表示其第  $i$  个元素。对于矩阵  $\mathbf{A}$ ,  $\mathbf{A}(i, j)$  为第  $i$  行  $j$  列元素。

对于给定的整数  $u \geq 3, 0 < m_1 < \dots < m_u$ , 算法 1 给出了  $u$  层的 MND, 其中第  $i$  层设计的试验次数为  $m_i$ 。在算法 1 及后文中,  $l_i$  表示  $m_1, \dots, m_i$  的最小公倍数。 $\mathbf{M}$  为一类循环矩阵, 对于给定的正整数  $n > m, \mathbf{M}_{m,n}$  为

$$\mathbf{M}_{m,n} = \begin{bmatrix} 1 & 2 & \dots & n \\ n & 1 & \dots & n-1 \\ \vdots & \vdots & \ddots & \vdots \\ m+1 & m+2 & \dots & m \end{bmatrix}$$

对于  $j = 1, \dots, u-1, \mathbf{v}_j$  为  $l_j$  维随机向量, 其中  $\mathbf{v}_j(1)$  服从  $Z_{l_j+1/l_j}$  上的离散均匀分布; 对于  $i = 2, \dots, l_j, \mathbf{v}_j(i) = \mathbf{M}_{l_j+1/m_{j+1}, l_j+1/l_j}(r_i, \mathbf{v}_j(i-1))$ , 其中  $r_i$  服从  $Z_{(l_j+1/l_j) - (l_j+1/m_{j+1}) + 1}$  上的离散均匀分布。

容易看出, 该算法有效需满足的条件是: 对于  $i = 1, \dots, u-1, m_{i+1}$  是  $m_i$  的倍数或  $m_{i+1} \geq l_i$ 。由于实际应用中, 高精度与低精度试验的试验点一般差距较大, 故大多数的试验可以满足上述两个条件。同时, 对于  $j = 1, \dots, u-1$ , 根据  $l_j$  维向量  $\mathbf{v}_j$  的定义可知, 其元素均属于  $Z_{l_j+1/l_j}$ 。再由  $\boldsymbol{\tau}_1$  的元素均属于  $Z_{m_1}$  可以得出  $\boldsymbol{\tau}_{j+1}$  的元素均属于  $Z_{l_{j+1}}$ , 因

### 算法 1 构造 MND

Alg. 1 Generate MND

输入: 嵌套矩阵的层数和各层的试验次数  
输出: MND

1. 向量  $\boldsymbol{\tau}_1$  是  $Z_{m_1}$  中元素的随机排列
2. **for**  $j$  从 1 到  $u-1$  **then**
3.     对于  $k = 1, \dots, m_j$
4.     **if**  $m_{j+1}$  是  $m_j$  的倍数 **then**
5.          $\boldsymbol{\tau}_{j+1}(k) = l_{j+1}(\boldsymbol{\tau}_j(k) - 1) / l_j + r_k$
6.          $r_k \in Z_{l_{j+1}/l_j}$  为随机数
7.     **elseif**  $m_{i+1} \geq l_i$  **then**
8.          $\boldsymbol{\tau}_{j+1}(k) = l_{j+1}(\boldsymbol{\tau}_j(k) - 1) / l_j + \mathbf{v}_j(\boldsymbol{\tau}_j(k))$
9.     **end if**
10.      $\boldsymbol{\tau}_{j+1}$  的后  $m_{i+1} - m_i$  个元素为集合  $\{r_i + l_{j+1}(s_i - 1) / m_{j+1}, i = 1, \dots, m_{j+1} - m_j\}$  的随机排列, 其中  $s_i$  是集合  $Z_{m_{j+1}} \setminus \{\lceil m_{j+1} \boldsymbol{\tau}_{j+1}(1) / l_{j+1} \rceil, \dots, \lceil m_{j+1} \boldsymbol{\tau}_{j+1}(m_j) / l_{j+1} \rceil\}$  以降序排列时的第  $i$  个元素,  $r_i \in Z_{l_{j+1}/m_{j+1}}$  为随机数。
11. **end for**
12. 令  $\mathbf{d}^{(1)} = (\boldsymbol{\tau}_u - \boldsymbol{\varepsilon}) / l_u$ , 其中  $m_u$  维向量  $\boldsymbol{\varepsilon}$  服从  $(0, 1]^{m_u}$  上的均匀分布。
13. 重复步骤 1~12 得到  $\mathbf{d}^{(2)}, \dots, \mathbf{d}^{(p)}$ 。
14. 则  $\mathbf{D} = (\mathbf{d}^{(1)}, \dots, \mathbf{d}^{(p)})$  为 MND。

此, 算法 1 步骤 8 中元素  $\mathbf{v}_j(\boldsymbol{\tau}_j(k))$  对任意的  $j = 1, \dots, u-1$  均存在。下面用一个例子来更好地说明该算法的流程。

**例 1** 对于  $u = 3, m_1 = 2, m_2 = 5, m_3 = 13, p = 1$ , 随机生成  $\boldsymbol{\tau}_1 = (2, 1)^T$ , 且  $\mathbf{v}_1$  来自矩阵  $\mathbf{M}_{2,5}$ 。随机构造后得向量  $\mathbf{v}_1 = (4, 2)^T$ , 则

$$\boldsymbol{\tau}_2(1) = 5(\boldsymbol{\tau}_1(1) - 1) + \mathbf{v}_1(\boldsymbol{\tau}_1(1)) = 7$$

同理,  $\boldsymbol{\tau}_2(2) = 4$ 。由算法 1 步骤 10 随机生成  $\boldsymbol{\tau}_2$  剩余的 3 个元素, 得  $\boldsymbol{\tau}_2 = (7, 4, 9, 2, 6)^T$ 。 $\mathbf{v}_2$  来自矩阵  $\mathbf{M}_{10,13}$ 。随机构造后得向量  $\mathbf{v}_2 = (1, 13, 10, 8, 6, 4, 2, 2, 12, 9)^T$ 。则

$$\boldsymbol{\tau}_3(1) = 13(\boldsymbol{\tau}_2(1) - 1) + \mathbf{v}_2(\boldsymbol{\tau}_2(1)) = 80$$

同理, 得到

$$\boldsymbol{\tau}_2 = (80, 47, 116, 26, 69, 124, 31, 95, 15, 90, 105, 60, 3)^T$$

则嵌套向量  $\mathbf{d} = (\boldsymbol{\tau}_3 - \boldsymbol{\varepsilon}) / 130$ 。图 1 显示了嵌套向量  $\mathbf{d}$ , 即 MND 的一维投影均匀性。由图可知, 对任一层, 其均可满足在任一区间中有且仅有一个点。

### 1.2 MND 的统计性质

进一步, 对于 MND 的性质有如下定理。

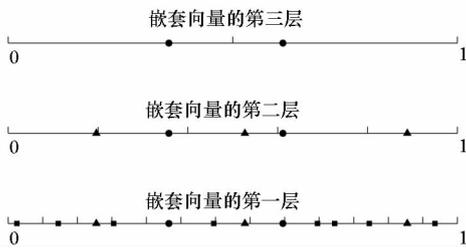


图 1 MND 的一维均匀性

Fig. 1 One-dimensional uniformity of MND

**定理 1** 对于给定的整数  $u \geq 3, 0 < m_1 < \dots < m_u, l$  为  $m_1, \dots, m_u$  的最小公倍数, 则嵌套向量  $\tau_u$  满足:

1) 对于任意的  $k = 1, \dots, u$  及  $i, j = 1, \dots, m_u$  且  $i \neq j$ , 有

$$\lceil \tau(i) m_k / l \rceil \neq \lceil \tau(j) m_k / l \rceil$$

2) 对于  $i = 1, \dots, m_u, a = 1, \dots, l$ , 有

$$P\{\tau_u(i) = a\} = 1 / (m_u l)$$

证明: 类似文献 [15] 中命题 1 的证明, 易知该定理成立。□

关于 MND 设计  $D = (d^{(1)}, \dots, d^{(p)})$  有如下两个定理。

**定理 2** 对于  $i = 1, \dots, u, j = 1, \dots, p, \{d^{(j)}(1), \dots, d^{(j)}(m_i)\}$  满足在任一区间  $[0, 1/m_i), \dots, [(m_i - 1)/m_i, 1)$  中有且仅有一个设计点。

证明: 由定理 1 第 1 部分可知。□

**定理 3** 对于  $i = 1, \dots, m_u, \{d^{(1)}(i), \dots, d^{(p)}(i)\}$  服从  $[0, 1)^p$  上的均匀分布。

证明: 由定理 1 第 2 部分可知。□

于是, 依据设计 MND 的估计量  $\hat{\mu}$  是  $\mu$  的无偏估计。

## 2 算例

对于运行次数分别为  $m_1, \dots, m_u$  的多种精度试验, 有四种方法可对其进行处理, 依次为:

1) 独立同分布抽样 (Independent and Identically Distributed sample, IID): 选取  $m_u$  行的独立同分布抽样, 利用其前  $m_i$  行处理第  $i$  个试验。

2) LHD: 选取  $m_u$  行的拉丁超立方体设计矩阵, 利用其前  $m_i$  行处理第  $i$  个试验。

3) 文献 [14] 的方法 SD。

## 4) 多层嵌套设计 MND。

令  $h_1(x), \dots, h_u(x)$  代表由高到低不同精度试验的响应函数,  $\hat{\mu}_k = (1/n) \sum_{x_i \in D} h_k(x)$  是在抽样  $D$  下响应函数  $h_k(x)$  的估计均值。利用估计的均方误差 (Root Mean Square Error, RMSE) 来对四种方法进行对比。两组函数来比较这四种方法的优劣:

1) M1:

$$h_1(x) = \log(1/\sqrt{x_1} + 1/\sqrt{x_2})$$

$$h_2(x) = \log(0.98/\sqrt{x_1} + 0.95/\sqrt{x_2})$$

$$h_3(x) = \log(1.02/\sqrt{x_1} + 1.02/\sqrt{x_2})$$

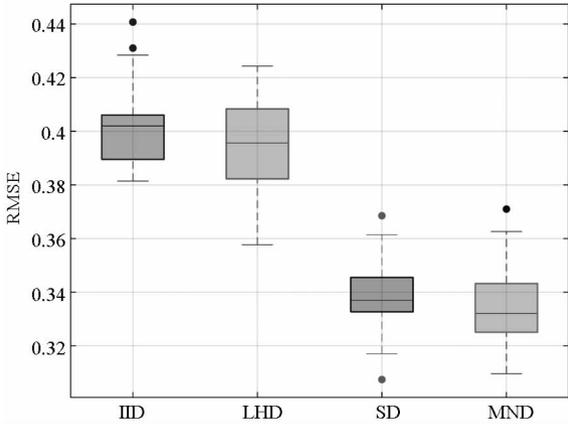
2) M2:

$$h_1(x) = \frac{2\pi x_3(x_4 - x_6)}{\log(x_2/x_1) \left[ 1 + \frac{2x_3x_7}{\log(x_2/x_1)x_1^2x_8} + x_3/x_5 \right]}$$

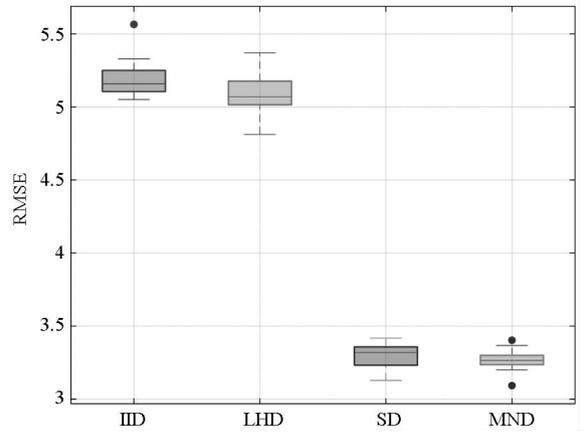
$$h_2(x) = \frac{5\pi x_3(x_4 - x_6)}{\log(x_2/x_1) \left[ 1.5 + \frac{2x_3x_7}{\log(x_2/x_1)x_1^2x_8} + x_3/x_5 \right]}$$

$$h_3(x) = \frac{7\pi x_3(x_4 - x_6)}{\log(x_2/x_1) \left[ 2.5 + \frac{3x_3x_7}{\log(x_2/x_1)x_1^2x_8} + x_3/x_5 \right]}$$

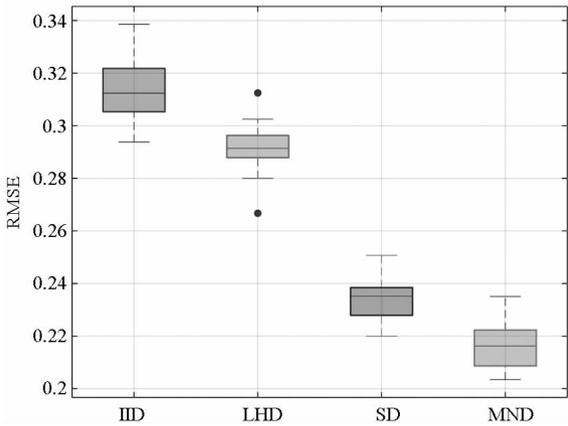
其中, M1 中的函数  $h_1(x)$  由文献 [16] 提出, M1 中  $h_2(x)$  和  $h_3(x)$  来自文献 [17]。M2 中  $h_1(x)$  被称为 borehole 函数<sup>[18]</sup>, 在计算机试验中经常选其作为黑箱函数<sup>[13,16,19]</sup>。在 M2 中代替低精度函数的  $h_2(x)$  和  $h_3(x)$  由文献 [13] 构造。对于不同方法生成的设计矩阵  $D$ , 其前  $m_1$  试验点对应模型  $h_1(x)$ , 前  $m_2$  试验点对应模型  $h_2(x)$ , 整个设计矩阵对应模型  $h_3(x)$ 。图 2 和图 3 分别为在 M1 和 M2 下不同方法估计  $\mu_i$  的均方误差。其中选定的参数为  $m_1 = 2, m_2 = 5, m_3 = 13$ , 每类试验进行 2000 次, 得出各方法的 RMSE 后进行对比。从两图中可以得到: ① IID 方法的均方误差最大; ② 在高精度试验中, SD 和 MND 均比 LHD 表现好, 其原因在于这两者的子矩阵比 LHD 的子矩阵有更好的均匀性; ③ 在估计  $\mu_1$  时, MND 与 SD 有几乎相同的均方误差; ④ 在估计  $\mu_2$  和  $\mu_3$  时, MND 比 SD 有更小的均方误差, 原因在于 MND 比 SD 在整体上有更好的均匀性; ⑤ 在估计  $\mu_3$  时, MND 和 LHD 有相同的 RMSE, 这是因为 MND 和 LHD 的全矩阵有相同的均匀性。



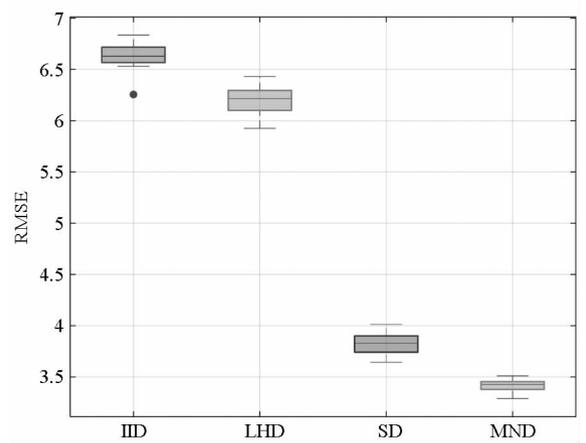
(a) M1 中模型  $h_1(x)$   
(a) Model  $h_1(x)$  in M1



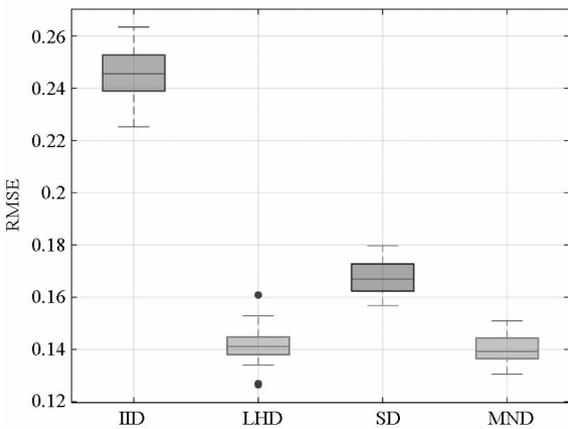
(a) M2 中模型  $h_1(x)$   
(a) Model  $h_1(x)$  in M2



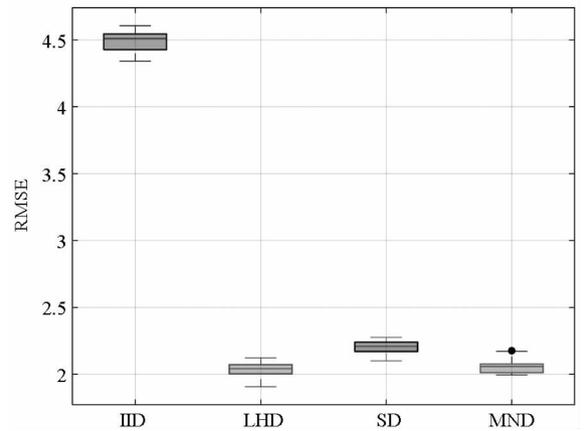
(b) M1 中模型  $h_2(x)$   
(b) Model  $h_2(x)$  in M1



(b) M2 中模型  $h_2(x)$   
(b) Model  $h_2(x)$  in M2



(c) M1 中模型  $h_3(x)$   
(c) Model  $h_3(x)$  in M1



(c) M2 中模型  $h_3(x)$   
(c) Model  $h_3(x)$  in M2

图 2 M1 下各方法 RMSE 比较

Fig. 2 RMSE comparison of the different schemes on M1

图 3 M2 下各方法 RMSE 比较

Fig. 3 RMSE comparison of the different schemes on M2

### 3 结论

本文构造了灵活的多层嵌套拉丁超立方体设计,与一般的嵌套设计相比,该方法在参数选取上具有很强的灵活性。与 SD 相比,该方法各层在

一维上均可达到更好的均匀性,这使得在估计模型均值时,其估计方差更小。两个算例也体现了多层嵌套拉丁超立方体设计的性质。虽然第 1 节中给出的算法也有其局限性,即对于  $i = 1, \dots,$

$u-1, m_{i+1}$  是  $m_i$  的倍数或  $m_{i+1} \geq l_i$  需满足, 但由于实际应用中, 高精度与低精度试验的试验点一般差距较大, 故大多数的试验可以满足上述两个条件。

## 参考文献 (References)

- [1] Santner T J, Williams B J, Notz W I. The design and analysis of computer experiments [M]. USA: Springer Science & Business Media, 2003.
- [2] Sacks J, Welch W J, Mitchell T J, et al. Design and analysis of computer experiments [J]. *Statistical Science*, 1989, 4(4): 205–206.
- [3] Fang K T, Li R Z, Sudjianto A. Design and modeling for computer experiments [M]. USA: Chapman and Hall/CRC Press, 2005.
- [4] Stein M. Large sample properties of simulations using Latin hypercube sampling [J]. *Technometrics*, 1987, 29 (2): 143–151.
- [5] McKay M D, Beckman R J, Conover W J. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code [J]. *Technometrics*, 1979, 21(2): 239–245.
- [6] Choi S, Alonso J J, Kroo I M, et al. Multifidelity design optimization of low-boom supersonic jets [J]. *Journal of Aircraft*, 2008, 45(1): 106–118.
- [7] Kennedy M C, O'Hagan A. Bayesian calibration of computer models [J]. *Journal of the Royal Statistical Society*, 2001, 63(3): 425–464.
- [8] Qian P Z G, Wu C F J. Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments [J]. *Technometrics*, 2008, 50(2): 192–204.
- [9] Reese C S, Wilson A G, Hamada M, et al. Integrated analysis of computer and physical experiments [J]. *Technometrics*, 2004, 46(2): 153–164.
- [10] Qian P Z G. Nested Latin hypercube designs [J]. *Biometrika*, 2009, 96(4): 957–970.
- [11] Qian P Z G, Wu H Q, Wu C F J. Gaussian process models for computer experiments with qualitative and quantitative factors [J]. *Technometrics*, 2008, 50(3): 383–396.
- [12] He X, Qian P Z G. Nested orthogonal array-based Latin hypercube designs [J]. *Biometrika*, 2011, 98 (3): 721–731.
- [13] Xiong S F, Qian P Z G, Wu C F J. Sequential design and analysis of high-accuracy and low-accuracy computer codes [J]. *Technometrics*, 2013, 55(1): 37–46.
- [14] Kong X S, Ai M Y, Tsui K L. Design for sequential follow-up experiments in computer emulations [J]. *Technometrics*, 2018, 60(1): 61–69.
- [15] Xu J, Duan X J, Wang Z M, et al. A general construction for nested Latin hypercube designs [J]. *Statistics & Probability Letters*, 2018, 134: 134–140.
- [16] Mease D, Bingham D. Latin hyperrectangle sampling for computer experiments [J]. *Technometrics*, 2006, 48 (4): 467–477.
- [17] Qian P Z G. Sliced Latin hypercube designs [J]. *Journal of the American Statistical Association*, 2012, 107 (497): 393–399.
- [18] Morris M D, Mitchell T J, Ylvisaker D. Bayesian design and analysis of computer experiments: use of derivatives in surface prediction [J]. *Technometrics*, 1993, 35(3): 243–255.
- [19] Xu J, Chen J J, Qian P Z G. Sequentially refined Latin hypercube designs: reusing every point [J]. *Journal of the American Statistical Association*, 2015, 110 (512): 1696–1706.