Journal of Statistical Planning and Inference 207 (2020) 113–122

Contents lists available at ScienceDirect

Journal of Statistical Planning and Inference

journal homepage: www.elsevier.com/locate/jspi

Optimal maximin L_2-distance Latin hypercube designs

Weiping Zhou, Jian-Feng Yang, Min-Qian Liu *

School of Statistics and Data Science, LPMC & KLMDASR, Nankai University, Tianjin 300071, China

ARTICLE INFO

Article history:
Received 14 March 2019
Received in revised form 14 September 2019
Accepted 22 November 2019
Available online 18 December 2019

MSC:
primary 62K05
secondary 62K15

Keywords:
Computer experiment
Orthogonality
Rotation method
Space-filling

ABSTRACT

Maximin distance Latin hypercube designs (LHDs) are extensively applied in computer experiments, but it is challenging to construct such designs. In this paper, based on a 2^2 full factorial design and a series of saturated two-level regular designs, a number of maximin distance LHDs are constructed via the rotation method. Some of the constructed LHDs are exactly optimal and the others are asymptotically optimal under the maximin L_2-distance criterion. The constructed maximin distance LHDs have two prominent advantages: (i) no computer search is needed; and (ii) they are orthogonal or nearly orthogonal. Detailed comparisons with existing LHDs show that the constructed LHDs have larger minimum distances between design points.

1. Introduction

In computer experiments, complex systems are increasingly investigated through space-filling designs, which aim to distribute the design points over the design space as evenly as possible. Latin hypercube designs (LHDs), first introduced by McKay et al. (1979), are used as a popular class of space-filling designs. As we know, LHDs achieve one-dimensional space-filling property. One disadvantage of LHDs is that any such design is not necessarily space-filling in the full-dimensional space. To solve this problem, maximin distance criterion (Johnson et al., 1990) was proposed for constructing good LHDs. The maximin distance criterion is to maximize the minimum distance between design points, which guarantees the good space-filling property in the full-dimensional space. The maximin distance designs are asymptotically optimal for fitting Gaussian process models under a Bayesian setting (Johnson et al., 1990), and the maximin distance LHDs are well-suited for computer experiments (Lin and Tang, 2015).

There are many algorithms for constructing maximin distance LHDs, such as the simulated annealing (Morris and Mitchell, 1995; Joseph and Hung, 2008; Ba et al., 2015), swarm optimization algorithms (Moon et al., 2011; Chen et al., 2013) and the threshold-accepting method (Xiao and Xu, 2018). However, due to the computational complexity, these methods are not suitable to construct large LHDs which are needed in computer experiments (see for example, Morris, 1991; Kleijnen, 1997; Cioppa and Lucas, 2007; Gramacy et al., 2015). In order to overcome the challenges for constructing large LHDs, Zhou and Xu (2015) considered linear permutations to construct maximin L_1- and L_2-distance LHDs based on good lattice point sets; Xiao and Xu (2017) constructed LHDs with large minimum L_1-distance via Costas arrays; Wang et al. (2018b) employed the Williams transformation to construct optimal maximin L_1-distance LHDs.

The rotation method, firstly presented by Beattie and Lin (2004, 2005), is simple and useful for constructing designs for computer experiments. This method was further employed to construct orthogonal LHDs, see e.g., Steinberg and Lin (2006), Lin et al. (2009), Pang et al. (2009), Sun and Tang (2017), and Wang et al. (2018a), among others. In this paper,

* Corresponding author.
E-mail address: mqliu@nankai.edu.cn (M.-Q. Liu).

https://doi.org/10.1016/j.jspi.2019.11.006

by combining the rotation method and the doubling operator of a design (Chen and Cheng, 2006), we propose several methods to construct maximin L_2-distance LHDs without any computer search. Firstly, based on a 2^2 full factorial design, a class of asymptotically optimal maximin L_2-distance LHDs are constructed via the rotation method. Moreover, we show that these LHDs are orthogonal. Next, based on a series of saturated two-level regular designs, a good deal of maximin L_2-distance LHDs are constructed via the rotation method. Some of these LHDs are exactly optimal and the others are asymptotically optimal under the maximin L_2-distance criterion. Furthermore, the average correlations of these LHDs converge to zero as the design sizes increase, which is desirable for Gaussian process with linear trend (Wang et al., 2018a,b).

The rest of this paper is organized as follows. Section 2 provides relevant notation and definitions. Section 3 presents the construction methods, along with some discussions of asymptotic properties for the L_2-distance efficiency of the resulting designs. Section 4 discusses several convergence properties of the average correlations for the resulting designs. Section 5 provides some concluding remarks. All proofs are deferred to Appendix.

2. Preliminaries

Throughout, $\mathbf{J}_{N\times n}$ is an $N \times n$ matrix of ones and $\mathbf{1}_k$ is a $k \times 1$ vector of ones. Let $\lfloor x \rfloor$ denote the integer part of x. Let $D(N, s^n)$ denote a design with N runs, n factors, and s levels, where each level occurs equally often in each factor. In this paper, an $N \times n$ matrix $L = (L_{ij})$ is called a Latin hypercube design (LHD), denoted by $L(N, n)$, when each column is a permutation of $-(N-1)/2, -(N-3)/2, \ldots, (N-3)/2, (N-1)/2$.

For any $N \times n$ design $D = (x_{ij})$, let $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$ be the ith row of D, define $d(\mathbf{x}_i, \mathbf{x}_k) = \sum_{j=1}^{n} (x_{ij} - x_{kj})^2$ as the L_2-distance of $\mathbf{x}_i$ and $\mathbf{x}_k$, and $d(D) = \min\{d(\mathbf{x}_i, \mathbf{x}_k) : i \neq k, i, k = 1, 2, \ldots, N\}$ as the L_2-distance of D. The maximin L_2-distance design is defined as the one which maximizes $d(D)$ (Johnson et al., 1990). Zhou and Xu (2015) derived the following upper bound of $d(D)$ for a $D(N, s^n)$ design D.

Lemma 1 (*Zhou and Xu, 2015*)**.**

(i) *For a $D(N, s^n)$ design D, $d(D) \leq \lfloor N(s^2-1)n/(6(N-1)) \rfloor$;*
(ii) *For any $N \times n$ LHD D, $d(D) \leq d_{\text{upper}} = \lfloor N(N+1)n/6 \rfloor$.*

From Lemma 1, for any $N \times n$ LHD D, define

$$d_{\text{eff}}(D) = d(D)/d_{\text{upper}} = d(D)/\lfloor N(N+1)n/6 \rfloor \quad (1)$$

as the L_2-distance efficiency of D. For any $N \times n$ design $D = (x_{ij})$, define

$$\rho_{\text{ave}}(D) = \frac{\sum_{j\neq k} |\rho_{jk}|}{n(n-1)}, \quad (2)$$

where ρ_{jk} denotes the correlation between the jth and kth columns of D. For any design D with entries from $\{0, 1\}$, let $\varphi_0(D) = D$ and

$$\varphi_k(D) = \begin{pmatrix} \varphi_{k-1}(D) & \varphi_{k-1}(D) \\ \varphi_{k-1}(D) & \varphi_{k-1}(D)+1 \end{pmatrix} \text{ for } k \geq 1,$$

where $\varphi_{k-1}(D) + 1$ is the matrix obtained by adding 1 (mod 2) to all the entries of $\varphi_{k-1}(D)$. Let

$$R_{10} = \begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix}, \quad R_{u0} = \begin{pmatrix} 2^{2^{(u-1)}} R_{(u-1)0} & -R_{(u-1)0} \\ R_{(u-1)0} & 2^{2^{(u-1)}} R_{(u-1)0} \end{pmatrix},$$

$$Q_1 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \text{ and } Q_u = \begin{pmatrix} Q_{u-1} & 0 \\ 0 & -Q_{u-1} \end{pmatrix},$$

for $u = 2, 3, \ldots$, then define

$$R_{u1} = \begin{pmatrix} 2R_{u0} & -Q_u \\ Q_u & 2R_{u0} \end{pmatrix} \text{ and } R_{uv} = \begin{pmatrix} 2R_{u(v-1)} & -Q_{u+v-1} \\ Q_{u+v-1} & 2R_{u(v-1)} \end{pmatrix} \text{ for } v = 2, 3, \ldots.$$

For $d = 2^u$, $u = 1, 2, \ldots$, if D with entries from $\{0, 1\}$ is a 2^d full factorial design, then $\left(\varphi_k(D) - (1/2)\mathbf{J}_{2^{d+k}\times 2^k d}\right) R_{uk}$ for $k \geq 0$ are the LHDs constructed by Sun and Tang (2017).

Lemma 2.

(i) *For $d = 2^u$, $u = 1, 2, \ldots$, let A be a 2^d full factorial design with entries from $\{0, 1\}$, and $A_k = \varphi_k(A)$ for $k \geq 1$. If $\mathbf{x}$ and $\mathbf{y}$ are two rows of A, then*

$$d((\mathbf{x} - (1/2)\mathbf{1}_d^T)R_{u0}, (\mathbf{y} - (1/2)\mathbf{1}_d^T)R_{u0}) = \frac{2^{2d}-1}{3} d(\mathbf{x}, \mathbf{y}),$$

Table 1
The LHDs F_0's in Example 1.

$m=1$		$m=2$				$m=3$					
−1.5	−0.5	−1.5	−0.5	−1.5	−0.5	−1.5	−0.5	−1.5	−0.5	−1.5	−0.5
−0.5	1.5	−0.5	1.5	0.5	−1.5	−0.5	1.5	0.5	−1.5	1.5	0.5
0.5	−1.5	0.5	−1.5	1.5	0.5	0.5	−1.5	1.5	0.5	−0.5	1.5
1.5	0.5	1.5	0.5	−0.5	1.5	1.5	0.5	−0.5	1.5	0.5	−1.5

and if $\mathbf{x}^k$ and $\mathbf{y}^k$ are two rows of A_k, then

$$d((\mathbf{x}^k-(1/2)\mathbf{1}^T_{2^kd})R_{uk},(\mathbf{y}^k-(1/2)\mathbf{1}^T_{2^kd})R_{uk})=\frac{2^{2(d+k)}-1}{3}d(\mathbf{x}^k,\mathbf{y}^k).$$

(ii) *Let E be an $N\times n$ matrix with entries from $\{0,1\}$, and $E_k=\varphi_k(E)$ for $k\geq 1$, then*

$$d(E_k)=2^{k-1}\min\{2d(E),n\}.$$

Lemma 2(i) tells us that the L_2-distance of the resulting design constructed via the rotation method is determined by that of the initial design, and Lemma 2(ii) shows that the L_2-distance of the large design $\varphi_k(E)$ is determined by that of the small design E. These findings are important in calculating the maximin distances of the constructed designs in the following sections.

3. Construction methods

In this section, we propose several methods for constructing maximin L_2-distance LHDs without any computer search. The following lemma is useful for the construction.

Lemma 3. *For $m=1,2,\ldots$, and $d=2^u$ with $u=1,2,\ldots$, let $E_0=(E_0^1,E_0^2,\ldots,E_0^m)$ be a $2^d\times md$ matrix, $F_0=(F_0^1,F_0^2,\ldots,F_0^m)$ be a $2^d\times md$ matrix and $F_k=(F_k^1,F_k^2,\ldots,F_k^m)$ be a $2^{d+k}\times 2^kmd$ matrix, where E_0^i is a 2^d full factorial design with entries from $\{0,1\}$, $F_0^i=(E_0^i-(1/2)\mathbf{J}_{2^d\times d})R_{u0}$ and $F_k^i=(\varphi_k(E_0^i)-(1/2)\mathbf{J}_{2^{d+k}\times 2^kd})R_{uk}$ for $k\geq 1$. Then*

$$d(F_0)=\frac{2^{2d}-1}{3}d(E_0),\ \ d_{eff}(F_0)\geq\frac{2d(E_0)}{md}(1-\frac{1}{2^d}),$$
$$d(F_k)=\frac{1}{3}(2^{2(d+k)}-1)2^{k-1}\min\{2d(E_0),md\}\text{ and }d_{eff}(F_k)\geq a(1-\frac{1}{2^{d+k}})\text{ for }k=1,2,\ldots,$$

where

$$a=\begin{cases}\frac{2d(E_0)}{md}, & \text{if}\quad 2d(E_0)<md;\\ 1, & \text{if}\quad 2d(E_0)\geq md.\end{cases}$$

It is worth noting that the choices of m in Lemma 3 are very broad. This makes it possible to generate many supersaturated LHDs. Obviously, the constructed LHDs F_k for $k\geq 0$ are supersaturated if $2^d\leq md$ in Lemma 3. The following example is an illustration for Lemma 3.

Example 1. Consider $m=1,2,3$ and $d=2$. Let $b_1=(0,0,1,1)^T$, $b_2=(0,1,0,1)^T$, $b_3=(0,1,1,0)^T$. Then (b_1,b_2,b_3) form a saturated 2^{3-1} regular design. For E_0, F_0 and F_1 in Lemma 3, it can be calculated that (i) for $m=1$, if $E_0=(b_1,b_2)$, then $d(E_0)=1$, $2d(E_0)=2$, $d(F_0)=5$ and $d(F_1)=42$; (ii) for $m=2$, if $E_0=(b_1,b_2,b_3,b_1)$, then $d(E_0)=2$, $2d(E_0)=4$, $d(F_0)=10$ and $d(F_1)=84$; (iii) for $m=3$, if $E_0=(b_1,b_2,b_3,b_1,b_2,b_3)$, then $d(E_0)=4$, $2d(E_0)=8>6$, $d(F_0)=20$ and $d(F_1)=126$, which all satisfy Lemma 3. The LHDs F_0's and F_1's are listed in Tables 1 and 2, respectively.

Lemma 3 shows that, for $i=0,1,\ldots$, the L_2-distance of F_i is determined by E_0, which means that F_i may be a good design when we choose E_0 with the largest L_2-distance. From Lemma 3, we can obtain that if $2d(E_0)\geq md$, then $d_{\mathrm{eff}}(F_k)$ converges to one as k tends to infinity; so F_k is asymptotical optimal under the maximin distance criterion. If A is a 2^2 full factorial design, then $d(A)=1$ which attains the upper bound of L_2-distance in Lemma 1(i). Let $E_0=A$, we can obtain the following result.

Theorem 1. *Suppose A is a 2^2 full factorial design with entries from $\{0,1\}$. For $k\geq 0$, let $L_k=(\varphi_k(A)-(1/2)\mathbf{J}_{2^{k+2}\times 2^{k+1}})R_{1k}$ be a $2^{k+2}\times 2^{k+1}$ matrix. Then $d(L_k)=2^k(2^{2(k+2)}-1)/3$ and $d_{\mathrm{eff}}(L_k)\geq 1-1/2^{k+2}$.*

Theorem 1 implies that $d(L_0)=5$ and $d_{\mathrm{eff}}(L_0)=5/6$. Also, it is easy to see from Theorem 1 that $d_{\mathrm{eff}}(L_k)$ converges to one as k tends to infinity. So L_k is asymptotical optimal under the maximin distance criterion. Table 3 compares the L_2-distances of L_k for $k=0,1,\ldots,10$ with that of the LHDs generated by the command *maximinSLHD* in the R package

Table 2
The LHDs F_1's in Example 1.

$m=1$					$m=2$							
−3.5	−0.5	−2.5	−1.5		−3.5	−0.5	−2.5	−1.5	−3.5	−0.5	−2.5	−1.5
−1.5	2.5	−0.5	3.5		−1.5	2.5	−0.5	3.5	1.5	−2.5	0.5	−3.5
1.5	−2.5	0.5	−3.5		1.5	−2.5	0.5	−3.5	3.5	0.5	2.5	1.5
3.5	0.5	2.5	1.5		3.5	0.5	2.5	1.5	−1.5	2.5	−0.5	3.5
−2.5	−1.5	3.5	0.5		−2.5	−1.5	3.5	0.5	−2.5	−1.5	3.5	0.5
−0.5	3.5	1.5	−2.5		−0.5	3.5	1.5	−2.5	0.5	−3.5	−1.5	2.5
0.5	−3.5	−1.5	2.5		0.5	−3.5	−1.5	2.5	2.5	1.5	−3.5	−0.5
2.5	1.5	−3.5	−0.5		2.5	1.5	−3.5	−0.5	−0.5	3.5	1.5	−2.5
$m=3$												
−3.5	−0.5	−2.5	−1.5	−3.5	−0.5	−2.5	−1.5	−3.5	−0.5	−2.5	−1.5	
−1.5	2.5	−0.5	3.5	1.5	−2.5	0.5	−3.5	3.5	0.5	2.5	1.5	
1.5	−2.5	0.5	−3.5	3.5	0.5	2.5	1.5	−1.5	2.5	−0.5	3.5	
3.5	0.5	2.5	1.5	−1.5	2.5	−0.5	3.5	1.5	−2.5	0.5	−3.5	
−2.5	−1.5	3.5	0.5	−2.5	−1.5	3.5	0.5	−2.5	−1.5	3.5	0.5	
−0.5	3.5	1.5	−2.5	0.5	−3.5	−1.5	2.5	2.5	1.5	−3.5	−0.5	
0.5	−3.5	−1.5	2.5	2.5	1.5	−3.5	−0.5	−0.5	3.5	1.5	−2.5	
2.5	1.5	−3.5	−0.5	−0.5	3.5	1.5	−2.5	0.5	−3.5	−1.5	2.5	

Table 3
Comparison of the L_2-distances for $2^{k+2} \times 2^{k+1}$ LHDs with $k \leq 10$.

k	L_k	SLHD		
		Min	Median	Max
0	5	5	5	5
1	42	28	31	42
2	340	236	264	285
3	2 728	1 983	2 195	2 331
4	21 840	16 881	18 252	18 718
5	174 752	141 884	149 918	152 597
6	1 398 080	1 218 585	1 227 107	1 227 804
7	11 184 768	9 782 026	9 812 963	9 870 116
8	89 478 400	67 362 353	68 900 044	69 360 506
9	715 827 712	589 692 664	590 153 127	594 475 277
10	5 726 622 720	4 911 137 878	4 944 641 146	5 025 068 011

Note: L_k: constructed by Theorem 1; SLHD: constructed by the R package SLHD.

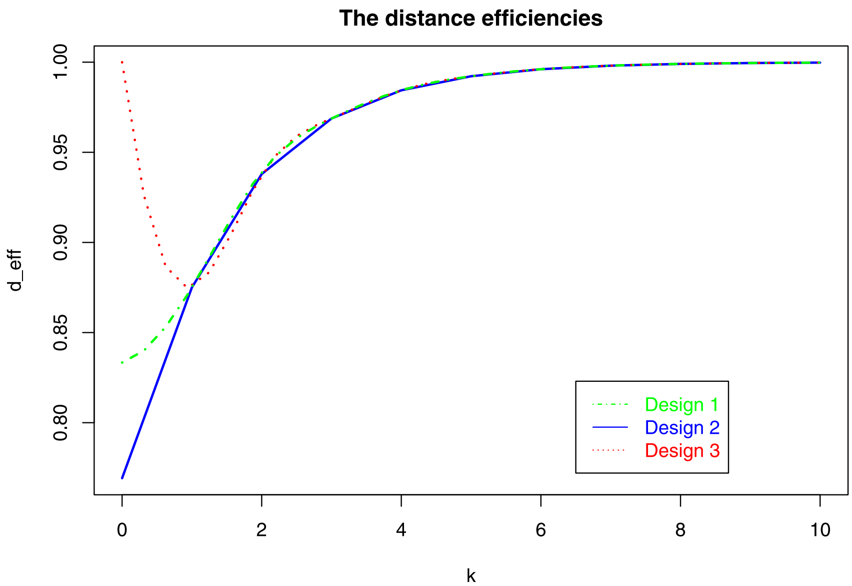

Fig. 1. Design 1: Theorem 1; Design 2: Algorithm 1 for $b=2$; Design 3: Algorithm 1 for $b=3$.

SLHD provided by Ba et al. (2015). Here, we ran the command repeatedly 100 times. From Table 3, when $k \geq 2$, L_k is better than SLHD under the maximin distance criterion. In Fig. 1, "Design 1" shows the values of $d_{\text{eff}}(L_k)$ for the L_k constructed by Theorem 1, where $k = 0, 1, \ldots, 10$. The $d_{\text{eff}}(L_k)$ increases fast as k increases and is greater than 0.9 when k is 2. When $k \geq 3$, the $d_{\text{eff}}(L_k)$ values are far greater than 0.95 and converge to 1.

It is worth noting that the constructed designs in Theorem 1 have the same L_2-distances with the designs constructed by Sun et al. (2009). Furthermore, by noting the existence of the mirror-symmetric structure, these designs can be shown to be optimal under the maximin L_2-distance criterion (Wang et al., 2018c).

Let $GF(2) = \{0, 1\}$ and $GF(2^d) = \{a_0 + a_1x + \cdots + a_{d-1}x^{d-1},\ a_i \in GF(2)\}$. It is worth noting that there exists a primitive polynomial $f(x)$ of degree d in $GF(2)$ such that each nonzero element of $GF(2^d)$ can be expressed as x^k modulo $f(x)$ in $GF(2^d)$ for $k \in \{0, 1, \ldots, 2^d - 2\}$. Let $\mathbf{1}, \mathbf{2}, \ldots, \mathbf{d}$ denote the d columns of a 2^d full factorial design. Each column, or a generated column, of $\mathbf{1}, \mathbf{2}, \ldots, \mathbf{d}$, can be expressed by $\mathbf{1}^{a_0}\mathbf{2}^{a_1}\cdots\mathbf{d}^{a_{d-1}}$ for some $a_i \in GF(2)$ and corresponds to a nonzero element $a_0 + a_1x + \cdots + a_{d-1}x^{d-1}$ of $GF(2^d)$. As indicated in Steinberg and Lin (2006), Pang et al. (2009) and Wang et al. (2018a), the corresponding columns of the nonzero elements of $GF(2^d)$, $x^0, x, \ldots, x^{2^d-2}$ modulo $f(x)$, form a saturated two-level regular design, denoted by B, and any d successive columns of B form a full factorial design. From Steinberg and Lin (2006), we have the following general result.

Lemma 4 (*Steinberg and Lin, 2006*)**.** *For any $t \geq 0$, the corresponding columns of the nonzero elements of $GF(2^d)$, $x^t, x^{t+1}, \ldots, x^{t+d-1}$ modulo $f(x)$, form a full factorial design.*

For the B defined above, $d(B) = 2^{d-1}$ (Mukerjee and Wu, 1995), which attains the upper bound of L_2-distance in Lemma 1(i). Lemmas 3 and 4 show that we can obtain optimal maximin L_2-distance LHDs based on this B. Next, we propose a new method for constructing maximin L_2-distance LHDs.

Algorithm 1.

Step 1. Given $d = 2^u$ for $u = 1, 2, \ldots$, obtain a saturated two-level regular design B as defined above, where B is a $2^d \times (2^d - 1)$ matrix.

Step 2. Let $q = \min\{g : g(2^d - 1) \pmod d) = 0,\ g = 1, 2, \ldots, d\}$. Let $C = \mathbf{1}_q^T \otimes B$. Write C as $C = (C_1, C_2, \ldots, C_\lambda)$, where $\lambda = q(2^d - 1)/d$ and C_i is a 2^d full factorial design.

Step 3. For $b = 1, 2, \ldots, \lambda$, and $k = 0, 1, \ldots$, let $L_k^b = (D_k^1, D_k^2, \ldots, D_k^b)$ be a $2^{d+k} \times 2^k bd$ design, where $D_k^i = (\varphi_k(C_i) - (1/2)\mathbf{J}_{2^{d+k}\times 2^k d})R_{uk}$ for $k \geq 0$.

Remark 1. Lemma 4 ensures that in Algorithm 1 the matrix C can be divided into λ groups of full factorial designs, and $b \leq \lambda$ ensures that there are no identical columns in L_k^b for $k = 0, 1, \ldots$.

Theorem 2. *Let $d = 2^u$ for $u = 1, 2, \ldots$. From Algorithm 1, we have that*

(i) *for $h_1 = 1, 2, \ldots, q - 1$, if $bd = h_1(2^d - 1) + h_2$ with $0 < h_2 < 2^d - 1$, then*

$$d_{\text{eff}}(L_0^b) \geq 1 - \frac{h_2}{h_1(2^d - 1) + h_2},\ \text{and}\ d_{\text{eff}}(L_k^b) \geq 1 - \frac{h_2 + 2^{-(d+k)}(2^d - 1)h_1}{h_1(2^d - 1) + h_2}\ \text{for}\ k \geq 1;$$

(ii) *if $bd = q(2^d - 1)$, i.e. $b = \lambda$, then L_0^b is a maximin L_2-distance LHD with $d(L_0^b) = 2^d(2^{2d} - 1)q/6$, and*

$$d_{\text{eff}}(L_k^b) \geq 1 - \frac{1}{2^{d+k}}\ \text{for}\ k \geq 1.$$

Theorem 2 shows that Algorithm 1 offers exact maximin L_2-distance LHDs, $L(2^2, 2(2^2-1))$, $L(2^4, 4(2^4-1))$, $L(2^8, 8(2^8-1))$, ... By noting that $d_{\text{eff}}(L_k^b)$ converges to one as k tends to infinity, Theorem 2 also shows that Algorithm 1 offers a class of asymptotically optimal maximin L_2-distance LHDs.

Example 2. Let $d = 2$. For the primitive polynomial $f(x) = x^2 + x + 1$ over $GF(2)$, we have $x^0 = 1$, $x = x$ and $x^2 = 1 + x$ over $GF(2^d)$. Thus we can obtain a saturated 2^{3-1} regular design $B = (b_1, b_2, b_3)$ with $b_1 = (0, 0, 1, 1)^T$, $b_2 = (0, 1, 0, 1)^T$ and $b_3 = (0, 1, 1, 0)^T$. From Algorithm 1, it is clear that $q = 2$, $C = (b_1, b_2, b_3, b_1, b_2, b_3)$ and $b = 1, 2, 3$. According to Algorithm 1, we can obtain LHDs L_k^i for $i = 2, 3$ and $k \geq 0$. Tables 4 and 5 compare the L_2-distances of L_k^i with that of the LHDs generated by the command *maximinSLHD* in R package SLHD provided by Ba et al. (2015). Here, we ran the command repeatedly 100 times. From Table 4, for $k \geq 2$, L_k^2 is better than SLHD under the maximin distance criterion. From Table 5, $d(L_0^3) = 20$, which attains the upper bound of L_2-distance in Lemma 1. For $k \geq 3$, L_k^3 is better than SLHD under the maximin distance criterion. In Fig. 1, "Design 2" and "Design 3" show the values of $d_{\text{eff}}(L_k^2)$ and $d_{\text{eff}}(L_k^3)$ respectively for $k = 0, 1, \ldots, 10$. It can be seen that both $d_{\text{eff}}(L_k^2)$ and $d_{\text{eff}}(L_k^3)$ increase fast as k increases and both are greater than 0.9 when k is 2. When $k \geq 3$, the $d_{\text{eff}}(L_k^2)$ and $d_{\text{eff}}(L_k^3)$ values are all far greater than 0.95 and converge to 1.

According to Theorem 1 and Algorithm 1, we can obtain a wealth of (asymptotically or exactly) optimal maximin L_2-distance LHDs. Table 6 presents a collection of optimal maximin L_2-distance LHDs of N runs and n factors with $N \leq 128$. In Table 6, the designs $L(N, n)$ with $n = N/2$ and $N = 4, 8, 16, 32, 64, 128$ are constructed using Theorem 1 and the others are constructed by Algorithm 1.

Table 4
Comparison of the L_2-distances for $2^{k+2} \times 2^{k+2}$ LHDs with $k \leq 10$.

k	L_k^2	SLHD		
		Min	Median	Max
0	10	12	12	12
1	84	78	84	88
2	680	605	647	676
3	5 456	4 796	5 098	5 296
4	43 680	38 759	40 808	41 978
5	349 504	321 028	330 454	332 708
6	2 796 160	2 630 278	2 630 971	2 637 670
7	22 369 536	20 631 324	20 644 054	20 674 515
8	178 956 800	164 035 301	162 892 810	163 182 053
9	1 431 655 424	1 252 525 609	1 260 334 711	1 263 607 947
10	11 453 245 440	10 336 809 965	10 357 207 925	10 430 594 387

Note: L_k^2: constructed by Algorithm 1; SLHD: constructed by the R package SLHD.

Table 5
Comparison of the L_2-distances for $2^{k+2} \times (6 \times 2^k)$ LHDs with $k \leq 10$.

k	L_k^3	SLHD		
		Min	Median	Max
0	20	20	20	20
1	126	120	132	136
2	1 020	948	1 011	1 035
3	8 184	7 644	7 967	8 123
4	65 520	63 168	63 860	64 344
5	524 256	508 802	509 944	511 015
6	4 194 240	4 013 245	4 057 468	4 065 083
7	33 554 304	31 613 152	31 707 607	31 736 751
8	268 435 200	228 219 327	229 885 890	232 327 045
9	2 147 483 136	1 921 440 435	1 924 776 449	1 933 439 986
10	17 179 868 160	15 856 028 901	15 869 758 518	15 916 222 828

Note: L_k^3: constructed by Algorithm 1; SLHD: constructed by the R package SLHD.

Table 6
Some optimal maximin L_2-distance LHDs of N runs and n factors with $N \leq 128$.

N	n	N	n	N	n	N	n	N	n	N	n	N	n
4	2	16	24	16[a]	60[a]	32	88	64	112	128	64	128	384
4	4	16	28	32	16	32	96	64	128	128	128	128	416
4[a]	6[a]	16	32	32	32	32	104	64	144	128	160	128	448
8	4	16	36	32	40	32	112	64	160	128	192	128	480
8	8	16	40	32	48	32	120	64	176	128	224		
8	12	16	44	32	56	64	32	64	192	128	256		
16	8	16	48	32	64	64	64	64	208	128	288		
16	16	16	52	32	72	64	80	64	224	128	320		
16	20	16	56	32	80	64	96	64	240	128	352		

[a]The exactly optimal maximin L_2-distance LHDs exist.

4. Orthogonality of the resulting designs

The $\rho_{\text{ave}}(D)$ measures the overall orthogonality of D. The design D with a small $\rho_{\text{ave}}(D)$ value is good for fitting the Gaussian process model with potential linear trend (Wang et al., 2018a,b). In this section, we consider the ρ_{ave} values of the LHDs with large L_2-distances constructed via Theorem 1 and Algorithm 1.

Proposition 1. *For the designs L_k constructed in Theorem 1, we have $\rho_{\text{ave}}(L_k) = 0$ for $k \geq 0$.*

Proposition 1 shows that orthogonal LHDs with large L_2-distances can be directly generated via Theorem 1 without any computer search. For the LHDs constructed by Algorithm 1, we have the following result.

Theorem 3. *Let $d = 2^u$ for $u = 1, 2, \ldots$. From Algorithm 1, we have that*

(i) *for $h_1 = 1, 2, \ldots, q-1$, if $bd = h_1(2^d - 1) + h_2$ with $0 < h_2 < 2^d - 1$, then*

$$\rho_{\text{ave}}(L_k^b) \leq \frac{3 \times (2^{d+k} - 1)(2^d - 1)(h_1 + 1)^2}{bd(2^k bd - 1)(2^{d+k} + 1)} - \frac{1}{2^k bd - 1} \text{ for } k \geq 0; \text{ and}$$

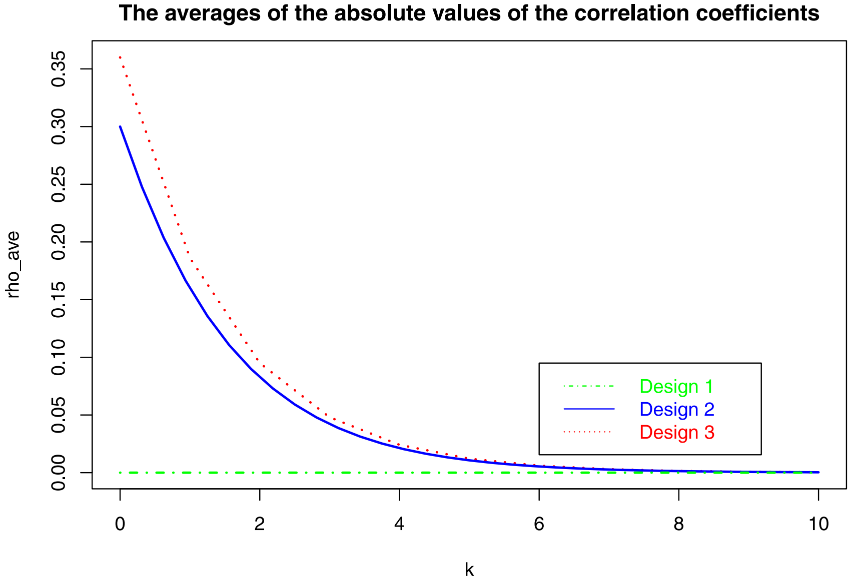

Fig. 2. Design 1: Theorem 1; Design 2: Algorithm 1 for $b = 2$; Design 3: Algorithm 1 for $b = 3$.

(ii) *if* $bd = q(2^d - 1)$*, i.e.* $b = \lambda$*, then*

$$\rho_{\text{ave}}(L_k^b) \leq \frac{3 \times (2^{d+k} - 1)(2^d - 1)q^2}{bd(2^k bd - 1)(2^{d+k} + 1)} - \frac{1}{2^k bd - 1} \text{ for } k \geq 0.$$

From Theorem 3, we can show that $\rho_{\text{ave}}(L_k^b)$ converges to zero as k tends to infinity. Thus, a class of LHDs with large L_2-distances and small ρ_{ave}'s can be easily generated via Algorithm 1 without any computer search. In Fig. 2, "Design 1", "Design 2" and "Design 3" show the values of $\rho_{\text{ave}}(L_k)$, $\rho_{\text{ave}}(L_k^2)$ and $\rho_{\text{ave}}(L_k^3)$ respectively, for $k = 0, 1, \ldots, 10$, where L_k is constructed by Theorem 1; L_k^2 and L_k^3 are constructed by Algorithm 1. Both $\rho_{\text{ave}}(L_k^2)$ and $\rho_{\text{ave}}(L_k^3)$ decrease fast as k increases and are less than 0.1 when k is 2. When $k \geq 3$, the $\rho_{\text{ave}}(L_k^2)$ and $\rho_{\text{ave}}(L_k^3)$ values are far less than 0.05 and converge to 0.

5. Concluding remarks

In this paper, we propose some methods for constructing maximin L_2-distance LHDs via the rotation method. The methods do not need any computer search and are more efficient especially for constructing large designs. They can lead to a class of asymptotically optimal maximin L_2-distance LHDs and exactly optimal maximin L_2-distance LHDs. Furthermore, some resulting designs are orthogonal and the average correlations of the other designs converge to zero as the design sizes increase.

The rotation method used in this paper has two major drawbacks. The first one is the limitation on the run size, which must be the power of two. If one can relax the requirement to work with LHDs, an alternative is to rotate non-regular two-level designs to generate nearly LHDs with flexible run sizes (Steinberg and Lin, 2015). Such designs are still desirable for many practical situations (Bingham et al., 2009; Sun et al., 2011; Ding et al., 2013; Jaynes et al., 2013). The second one is the low coverage in low dimensional subspaces. The new factors in the resulting design naturally divide into pairs, where each pair has the two largest rotation weights on the same original factors in the two-level design. In the projection onto these two factors, all the design points concentrate in just a few cells of a coarser binary grid. In order to overcome this drawback, Steinberg and Lin (2015) recommended choosing just one factor from each of such pairs. This will improve the two-dimensional coverage a lot, although the optimality under the maximin distance criterion cannot be guaranteed any more.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Nos. 11771220, 11871033, 11431006, 11671386 and 11971204), National Ten Thousand Talents Program, China, Tianjin Development Program for Innovation and Entrepreneurship, China, Tianjin "131" Talents Program, China, and Project 61331903. The first two authors contributed equally to this work.

Appendix. Proofs

Proof of Lemma 2. (i) Let $\mathbf{x} = (x_1, \ldots, x_d)$ and $\mathbf{y} = (y_1, \ldots, y_d)$ be two rows of A. It is clear that

$$d((\mathbf{x} - (1/2)\mathbf{1}_d^T)R_{u0}, (\mathbf{y} - (1/2)\mathbf{1}_d^T)R_{u0}) = (\mathbf{x} - \mathbf{y})R_{u0}R_{u0}^T(\mathbf{x} - \mathbf{y})^T = \frac{2^{2d} - 1}{3} d(\mathbf{x}, \mathbf{y}),$$

since $R_{u0}R_{u0}^T = (2^{2d} - 1)/3\mathbf{I}_d$. Similarly,

$$d((\mathbf{x}^k - (1/2)\mathbf{1}_{2^k d}^T)R_{uk}, (\mathbf{y}^k - (1/2)\mathbf{1}_{2^k d}^T)R_{uk}) = \frac{2^{2(d+k)} - 1}{3} d(\mathbf{x}^k, \mathbf{y}^k),$$

since $R_{uk}R_{uk}^T = (2^{2(d+k)} - 1)/3\mathbf{I}_{2^k d}$.

(ii) Let $\mathbf{x}$ and $\mathbf{y}$ be two rows of E, then $(\mathbf{x}, \mathbf{x})$, $(\mathbf{x}, \mathbf{x} + 1 (\mathrm{mod}\ 2))$, $(\mathbf{y}, \mathbf{y})$, $(\mathbf{y}, \mathbf{y} + 1 (\mathrm{mod}\ 2))$ are four rows of E_1. It is clear that $d((\mathbf{x}, \mathbf{x}), (\mathbf{y}, \mathbf{y})) = d((\mathbf{x}, \mathbf{x} + 1 (\mathrm{mod}\ 2)), (\mathbf{y}, \mathbf{y} + 1 (\mathrm{mod}\ 2))) = 2d(\mathbf{x}, \mathbf{y})$ and $d((\mathbf{x}, \mathbf{x}), (\mathbf{x}, \mathbf{x} + 1 (\mathrm{mod}\ 2))) = d((\mathbf{y}, \mathbf{y}), (\mathbf{y}, \mathbf{y} + 1 (\mathrm{mod}\ 2))) = d((\mathbf{x}, \mathbf{x}), (\mathbf{y}, \mathbf{y} + 1 (\mathrm{mod}\ 2))) = d((\mathbf{x}, \mathbf{x} + 1 (\mathrm{mod}\ 2)), (\mathbf{y}, \mathbf{y})) = n$. Thus $d(E_1) = \min\{2d(E), n\}$. For $k > 1$, by recursion we can obtain $d(E_k) = 2^{k-1}\min\{2d(E), n\}$. This completes the proof. □

Proof of Lemma 3. From Lemma 2, it is clear that Lemma 3 is true. □

Proof of Theorem 1. The results just follow from Lemma 3. □

Proof of Theorem 2. Let $C_k^{(b)} = (\varphi_k(C_1), \ldots, \varphi_k(C_b))$ and $B_k = \varphi_k(B)$ with $k \geq 0$. It is clear from Lemma 1(ii) that $d_{\text{upper}} = \lfloor 2^{d+k}(2^{d+k} + 1)2^k bd/6 \rfloor$ for the L_k^b with $k \geq 0$.

(i) From Lemma 2(i), it is known that

$$d(C_0^{(b)}) \geq h_1 d(B) = 2^{d-1}h_1 \quad \text{and} \quad d(L_0^b) = \frac{2^{2d} - 1}{3} d(C_0^{(b)}) \geq 2^{d-1}\frac{2^{2d} - 1}{3} h_1.$$

From Lemma 2(ii), $d(B_k) = 2^{k-1}(2^d - 1)$ for $k \geq 1$. We can obtain that

$$d(L_k^b) = \frac{2^{2(d+k)} - 1}{3} d(C_k^{(b)}) \geq \frac{(2^{2(d+k)} - 1)h_1}{3} d(B_k),$$

since $d(C_k^{(b)}) \geq h_1 d(B_k)$ for $k \geq 1$. Thus Theorem 2(i) is true from (1).

(ii) If $bd = q(2^d - 1)$, then

$$d(C_0^{(b)}) = qd(B) = q2^{d-1} \text{ and } d(L_0^b) = \frac{2^{2d} - 1}{3} d(C_0^{(b)}) = \frac{2^d(2^{2d} - 1)q}{6}.$$

From Lemma 1, it is known that L_0^b is a maximin distance LHD. Furthermore, it is clear that for $k \geq 1$,

$$d(L_k^b) = \frac{2^{2(d+k)} - 1}{3} d(C_k^{(b)}) = \frac{(2^{2(d+k)} - 1)q}{3} d(B_k) = \frac{2^{k-1}(2^d - 1)(2^{2(d+k)} - 1)q}{3}.$$

Thus Theorem 2(ii) is true from (1). This completes the proof. □

Let $D = (x_{ij})$ be an $N \times n$ matrix, define $\mathrm{Sum}(D) = \sum_{i=1}^N \sum_{j=1}^n x_{ij}$, and $\mathrm{Abs}(D) = (|x_{ij}|)$ where $|x_{ij}|$ is the absolute value of x_{ij}. To prove Theorem 3, the following lemma is crucial.

Lemma 5. *In Algorithm 1, let* $M_0^{(b)} = (C_1 - (1/2)\mathbf{J}_{2^d \times d}, \ldots, C_b - (1/2)\mathbf{J}_{2^d \times d})$, $M_k^{(b)} = (\varphi_k(C_1) - (1/2)\mathbf{J}_{2^{d+k} \times 2^k d}, \ldots, \varphi_k(C_b) - (1/2)\mathbf{J}_{2^{d+k} \times 2^k d})$ *with* $k \geq 1$. *We have that*

(i) *for* $h_1 = 1, 2, \ldots, q - 1$, *if* $bd = h_1(2^d - 1) + h_2$ *with* $0 < h_2 < 2^d - 1$, *then* $\mathrm{Sum}(\mathrm{Abs}((M_k^{(b)})^T M_k^{(b)})) \leq 4^{k-1}2^d(2^d - 1)(h_1 + 1)^2$ *for* $k = 0, 1, \ldots$;

(ii) *if* $bd = q(2^d - 1)$, *then* $\mathrm{Sum}(\mathrm{Abs}((M_k^{(b)})^T M_k^{(b)})) = 4^{k-1}2^d(2^d - 1)q^2$ *for* $k = 0, 1, \ldots$.

Proof of Lemma 5. For the B in Algorithm 1, $(B - (1/2)\mathbf{J}_{2^d \times (2^d - 1)})^T(B - (1/2)\mathbf{J}_{2^d \times (2^d - 1)}) = 2^{d-2}\mathbf{I}_{2^d - 1}$. For $k = 1, 2, \ldots$, let $M_k^{(b)} = (M_k^{(b1)}, \ldots, M_k^{(bb)})$ with $M_k^{(bi)} = \varphi_k(C_i) - (1/2)\mathbf{J}_{2^{d+k} \times 2^k d}$, then

$$M_k^{(bi)} = \begin{pmatrix} M_{k-1}^{(bi)} & M_{k-1}^{(bi)} \\ M_{k-1}^{(bi)} & -M_{k-1}^{(bi)} \end{pmatrix} \text{ and } (M_k^{(bi)})^T M_k^{(bj)} = \begin{pmatrix} 2(M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} & 0 \\ 0 & 2(M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} \end{pmatrix}.$$

Thus

$$\mathrm{Sum}(\mathrm{Abs}((M_k^{(bi)})^T M_k^{(bj)})) = 4\mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})) \text{ and } \mathrm{Sum}(\mathrm{Abs}((M_k^{(b)})^T M_k^{(b)})) = 4\mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(b)})^T M_{k-1}^{(b)})).$$

Let $M_0^{(b)*} = (\mathbf{1}_{(h_1+1)}^T \otimes B) - (1/2)\mathbf{J}_{2^d\times(h_1+1)(2^d-1)}$.

(i) If $bd = h_1(2^d-1)+h_2$ with $0 < h_2 < 2^d-1$, then for $k \ge 1$,
$\mathrm{Sum}(\mathrm{Abs}((M_0^{(b)})^T M_0^{(b)})) \le \mathrm{Sum}(\mathrm{Abs}((M_0^{(b)*})^T M_0^{(b)*})) = 2^{d-2}(2^d-1)(h_1+1)^2$, and
$\mathrm{Sum}(\mathrm{Abs}((M_k^{(b)})^T M_k^{(b)})) = 4^k\mathrm{Sum}(\mathrm{Abs}((M_0^{(b)})^T M_0^{(b)})) \le 4^{k-1}2^d(2^d-1)(h_1+1)^2$.

(ii) If $bd = q(2^d-1)$, then $M_0^{(b)} = C - (1/2)\mathbf{J}_{2^d\times q(2^d-1)} = (\mathbf{1}_q^T \otimes B) - (1/2)\mathbf{J}_{2^d\times q(2^d-1)}$, and
$\mathrm{Sum}(\mathrm{Abs}((M_0^{(b)})^T M_0^{(b)})) = 2^{d-2}(2^d-1)q^2$. Thus
$\mathrm{Sum}(\mathrm{Abs}((M_k^{(b)})^T M_k^{(b)})) = 4^k\mathrm{Sum}(\mathrm{Abs}((M_0^{(b)})^T M_0^{(b)})) = 4^{k-1}2^d(2^d-1)q^2$ for $k \ge 1$.

This completes the proof. □

Proof of Theorem 3. Let $M_0^{(b)} = (M_0^{(b1)}, \ldots, M_0^{(bb)})$ with $M_0^{(bi)} = C_i - (1/2)\mathbf{J}_{2^d\times d}$. For $k = 1, 2, \ldots$, let $M_k^{(b)} = (M_k^{(b1)}, \ldots, M_k^{(bb)})$ with $M_k^{(bi)} = \varphi_k(C_i) - (1/2)\mathbf{J}_{2^{d+k}\times 2^k d}$.

For $i = 1, 2, \ldots, b$, $j = 1, 2, \ldots, b$,

$$\begin{aligned}
\mathrm{Sum}(\mathrm{Abs}((D_0^i)^T D_0^j)) &= \mathrm{Sum}(\mathrm{Abs}(R_{u0}^T (M_0^{(bi)})^T M_0^{(bj)} R_{u0})) \\
&\le \left(\textstyle\sum_{h=0}^{2^u-1} 2^h\right)^2 \mathrm{Sum}(\mathrm{Abs}((M_0^{(bi)})^T M_0^{(bj)})) \\
&= (2^d-1)^2\mathrm{Sum}(\mathrm{Abs}((M_0^{(bi)})^T M_0^{(bj)})), \text{ and} \\
\mathrm{Sum}(\mathrm{Abs}((L_0^b)^T L_0^b)) &\le (2^d-1)^2\mathrm{Sum}(\mathrm{Abs}((M_0^{(b)})^T M_0^{(b)})).
\end{aligned}$$

From Lemma 5, we have that for $h_1 = 1, 2, \ldots, q-1$, if $bd = h_1(2^d-1)+h_2$ with $0 < h_2 < 2^d-1$, then $\mathrm{Sum}(\mathrm{Abs}((L_0^b)^T L_0^b)) \le 2^{d-2}(2^d-1)^3(h_1+1)^2$; and if $bd = q(2^d-1)$, then $\mathrm{Sum}(\mathrm{Abs}((L_0^b)^T L_0^b)) \le 2^{d-2}(2^d-1)^3 q^2$. It is clear that

$$\rho_{\mathrm{ave}}(L_0^b) = \frac{1}{bd(bd-1)}\left(\frac{12}{2^d(2^{2d}-1)}\mathrm{Sum}(\mathrm{Abs}((L_0^b)^T L_0^b)) - bd\right).$$

Thus for $h_1 = 1, 2, \ldots, q-1$, if $bd = h_1(2^d-1)+h_2$ with $0 < h_2 < 2^d-1$, then

$$\rho_{\mathrm{ave}}(L_0^b) \le \frac{3(2^d-1)^2(h_1+1)^2}{bd(bd-1)(2^d+1)} - \frac{1}{bd-1};$$

if $bd = q(2^d-1)$, then

$$\rho_{\mathrm{ave}}(L_0^b) \le \frac{3q(2^d-1)}{(q(2^d-1)-1)(2^d+1)} - \frac{1}{q(2^d-1)-1}.$$

For $k = 1, 2, \ldots$, $(L_k^b)^T L_k^b = \left((D_k^i)^T D_k^j\right)_{i,j=1,2,\ldots,b}$, where $(D_k^i)^T D_k^j = R_{uk}^T (M_k^{(bi)})^T M_k^{(bj)} R_{uk}$. Thus

$$(D_k^i)^T D_k^j = \begin{pmatrix} 8U_{k-1}^{(bij)} + 2V_{k-1}^{(bij)} & -4W_{k-1}^{(bij)} + 4Z_{k-1}^{(bij)} \\ 4W_{k-1}^{(bij)} - 4Z_{k-1}^{(bij)} & 8U_{k-1}^{(bij)} + 2V_{k-1}^{(bij)} \end{pmatrix},$$

where $U_{k-1}^{(bij)} = R_{u(k-1)}^T (M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} R_{u(k-1)}$, $V_{k-1}^{(bij)} = Q_{u+k-1}^T (M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} Q_{u+k-1}$, $W_{k-1}^{(bij)} = R_{u(k-1)}^T (M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} Q_{u+k-1}$, and $Z_{k-1}^{(bij)} = Q_{u+k-1}^T (M_{k-1}^{(bi)})^T M_{k-1}^{(bj)} R_{u(k-1)}$. We can obtain that

$$\begin{aligned}
\mathrm{Sum}(\mathrm{Abs}(U_{k-1}^{(bij)})) &\le \left(\textstyle\sum_{h=0}^{2^u+k-2} 2^h\right)^2 \mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})), \\
\mathrm{Sum}(\mathrm{Abs}(V_{k-1}^{(bij)})) &= \mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})), \\
\mathrm{Sum}(\mathrm{Abs}(W_{k-1}^{(bij)})) &\le \left(\textstyle\sum_{h=0}^{2^u+k-2} 2^h\right) \mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})), \text{ and} \\
\mathrm{Sum}(\mathrm{Abs}(Z_{k-1}^{(bij)})) &\le \left(\textstyle\sum_{h=0}^{2^u+k-2} 2^h\right) \mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})).
\end{aligned}$$

Thus

$$\begin{aligned}
\mathrm{Sum}(\mathrm{Abs}((D_k^i)^T D_k^j)) &\le 4(2^{d+k}-1)^2\mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(bi)})^T M_{k-1}^{(bj)})), \text{ and} \\
\mathrm{Sum}(\mathrm{Abs}((L_k^b)^T L_k^b)) &\le 4(2^{d+k}-1)^2\mathrm{Sum}(\mathrm{Abs}((M_{k-1}^{(b)})^T M_{k-1}^{(b)})).
\end{aligned}$$

It is clear that

$$\begin{aligned}\rho_{\text{ave}}(L_k^b) &= \frac{1}{2^k bd(2^k bd-1)}\left(\frac{12}{2^{d+k}(2^{2(d+k)}-1)}\text{Sum}(\text{Abs}((L_k^b)^T L_k^b)) - 2^k bd\right)\\ &\le \frac{1}{2^k bd(2^k bd-1)}\left(\frac{12}{2^{d+k}(2^{2(d+k)}-1)}4(2^{d+k}-1)^2\text{Sum}(\text{Abs}((M_{k-1}^{(b)})^T M_{k-1}^{(b)})) - 2^k bd\right).\end{aligned}$$

From Lemma 5, it is known that for $k \ge 1$ and $h_1 = 1, 2, \ldots, q-1$, if $bd = h_1(2^d-1)+h_2$ with $0 < h_2 < 2^d - 1$, then

$$\begin{aligned}\rho_{\text{ave}}(L_k^b) &\le \frac{1}{2^k bd(2^k bd-1)}\left(\frac{12\times 4^{k-1}(2^{d+k}-1)^2 2^d(2^d-1)(h_1+1)^2}{2^{d+k}(2^{2(d+k)}-1)} - 2^k bd\right)\\ &= \frac{3\times(2^{d+k}-1)(2^d-1)(h_1+1)^2}{bd(2^k bd-1)(2^{d+k}+1)} - \frac{1}{2^k bd-1},\end{aligned}$$

and if $bd = q(2^d-1)$, then

$$\begin{aligned}\rho_{\text{ave}}(L_k^b) &\le \frac{1}{2^k bd(2^k bd-1)}\left(\frac{12\times 4^{k-1}(2^{d+k}-1)^2 2^d(2^d-1)q^2}{2^{d+k}(2^{2(d+k)}-1)} - 2^k bd\right)\\ &= \frac{3\times(2^{d+k}-1)(2^d-1)q^2}{bd(2^k bd-1)(2^{d+k}+1)} - \frac{1}{2^k bd-1}.\end{aligned}$$

This completes the proof. □

References

Ba, S., Myers, W.R., Brenneman, W.A., 2015. Optimal sliced Latin hypercube designs. Technometrics 57, 479–487.
Beattie, S.D., Lin, D.K.J., 2004. Rotated factorial designs for computer experiments. J. Chin. Stat. Assoc. 42, 431–450.
Beattie, S.D., Lin, D.K.J., 2005. A new class of Latin hypercube for computer experiments. In: Fan, J., Li, G. (Eds.), Contemporary Multivariate Analysis and Experimental Designs in Celebration of Professor Kai-Tai Fang'S 65th Birthday. World Scientific, Singapore, pp. 205–226.
Bingham, D., Sitter, R.R., Tang, B., 2009. Orthogonal and nearly orthogonal designs for computer experiments. Biometrika 96, 51–65.
Chen, H., Cheng, C.S., 2006. Doubling and projection: A method of constructing two-level designs of resolution IV. Ann. Statist. 34, 546–558.
Chen, R.B., Hsieh, D.N., Hung, Y., Wang, W., 2013. Optimizing Latin hypercube designs by particle swarm. Statist. Comput. 23, 663–676.
Cioppa, T.M., Lucas, T.W., 2007. Efficient nearly orthogonal and space-filling Latin hypercubes. Technometrics 49, 45–55.
Ding, X., Xu, H., Hopper, C., Yang, J., Ho, C.M., 2013. Use of fractional factorial designs in antiviral drug studies. Qual. Reliab. Eng. Int. 29, 299–304.
Gramacy, R.B., Bingham, D., Holloway, J.P., Grosskopf, M.J., Kuranz, C.C., Rutter, E., Trantham, M., Drake, R.P., 2015. Calibrating a large computer experiment simulating radiative shock hydrodynamics. Ann. Appl. Stat. 9, 1141–1168.
Jaynes, J., Ding, X., Xu, H., Wong, W.K., Ho, C.M., 2013. Application of fractional factorial designs to study drug combinations. Stat. Med. 32, 307–318.
Johnson, M.E., Moore, L.M., Ylvisaker, D., 1990. Minimax and maximin distance designs. J. Statist. Plann. Inference 26, 131–148.
Joseph, V.R., Hung, Y., 2008. Orthogonal maximin Latin hypercube designs. Statist. Sinica 18, 171–186.
Kleijnen, J.P., 1997. Sensitivity analysis and related analyses: a review of some statistical techniques. J. Stat. Comput. Simul. 57, 111–142.
Lin, C.D., Mukerjee, R., Tang, B., 2009. Construction of orthogonal and nearly orthogonal Latin hypercubes. Biometrika 96, 243–247.
Lin, C.D., Tang, B., 2015. Latin hypercubes and space-filling designs. In: Dean, A., Morris, M., Stufken, J., Bingham, D. (Eds.), Handbook of Design and Analysis of Experiments. Chapman and Hall/CRC, Boca Raton, pp. 593–625.
McKay, M.D., Beckman, R.J., Conover, W.J., 1979. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21, 239–245.
Moon, H., Dean, A., Santner, T., 2011. Algorithms for generating maximin Latin hypercube and orthogonal designs. J. Stat. Theory Pract. 5, 81–98.
Morris, M.D., 1991. Factorial plans for preliminary computational experiments. Technometrics 33, 161–174.
Morris, M.D., Mitchell, T.J., 1995. Exploratory designs for computational experiments. J. Statist. Plann. Inference 43, 381–402.
Mukerjee, R., Wu, C.F.J., 1995. On the existence of saturated and nearly saturated asymmetrical orthogonal arrays. Ann. Statist. 23, 2102–2115.
Pang, F., Liu, M.Q., Lin, D.K.J., 2009. A construction method for orthogonal Latin hypercube designs with prime power levels. Statist. Sinica 19, 1721–1728.
Steinberg, D.M., Lin, D.K.J., 2006. A construction method for orthogonal Latin hypercube designs. Biometrika 93, 279–288.
Steinberg, D.M., Lin, D.K.J., 2015. Construction of orthogonal nearly Latin hypercubes. Qual. Reliab. Eng. Int. 31, 1397–1406.
Sun, F.S., Liu, M.Q., Lin, D.K.J., 2009. Construction of orthogonal Latin hypercube designs. Biometrika 96, 971–974.
Sun, F.S., Pang, F., Liu, M.Q., 2011. Construction of column-orthogonal designs for computer experiments. Sci. China Math. 54, 2683–2692.
Sun, F.S., Tang, B., 2017. A general rotation method for orthogonal Latin hypercubes. Biometrika 104, 465–472.
Wang, L., Sun, F.S., Lin, D.K.J., Liu, M.Q., 2018a. Construction of orthogonal symmetric Latin hypercube designs. Statist. Sinica 28, 1503–1520.
Wang, L., Xiao, Q., Xu, H., 2018b. Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. Ann. Statist. 46, 3741–3766.
Wang, Y.P., Yang, J.F., Xu, H., 2018c. On the connection between maximin distance designs and orthogonal designs. Biometrika 105, 471–477.
Xiao, Q., Xu, H., 2017. Construction of maximin distance Latin squares and related Latin hypercube designs. Biometrika 104, 455–464.
Xiao, Q., Xu, H., 2018. Construction of maximin distance designs via level permutation and expansion. Statist. Sinica 28, 1395–1414.
Zhou, Y.D., Xu, H., 2015. Space-filling properties of good lattice point sets. Biometrika 102, 959–966.