Biometrika (2021), **108**, 4, *pp.* 845–855
Printed in Great Britain
doi: 10.1093/biomet/asaa089
Advance Access publication 12 August 2021

A method of constructing maximin distance designs

BY WENLONG LI, MIN-QIAN LIU

School of Statistics and Data Science, Nankai University,
94 Weijin Road, Tianjin 300071, China
wenlongli@mail.nankai.edu.cn mqliu@nankai.edu.cn

AND BOXIN TANG

Department of Statistics and Actuarial Science, Simon Fraser University,
Burnaby, British Columbia V5A 1S6, Canada
boxint@sfu.ca

SUMMARY

An attractive type of space-filling design for computer experiments is the class of maximin distance designs. Algorithmic search is commonly used for finding such designs, but this approach becomes ineffective for large problems. Theoretical construction of maximin distance designs is challenging; some results have been obtained recently, often using highly specialized techniques. This article presents an easy-to-use method for constructing maximin distance designs. The method is versatile as it works with any distance measure. The basic idea is to construct large designs from small designs, and the method is effective because the quality of large designs is guaranteed by that of small designs, as evaluated by the maximin distance criterion.

Some key words: Computer experiment; Orthogonal array; Space-filling design.

1. INTRODUCTION

Computer experiments are powerful tools for investigating complex systems in the sciences and engineering. The most commonly used designs for computer experiments are space-filling designs (Santner et al., 2003; Fang et al., 2006), which aim to distribute the design points over the design space as uniformly as possible. One can construct designs that are space-filling in low dimensions by using orthogonal arrays or stronger versions of such arrays. Research in this area originated from the work of McKay et al. (1979), was continued by Owen (1992) and Tang (1993), and remains very active to the present day. For some recent developments, see Mukerjee et al. (2014) and He et al. (2018).

Maximin distance designs, first introduced by Johnson et al. (1990), are also popular choices in the design of computer experiments. Johnson et al. (1990) showed that maximin distance designs are asymptotically optimal in a Bayesian setting when Gaussian process models are considered. But finding maximin distance designs is no simple matter. One may resort to algorithmic search as in Ba (2013). An algorithmic search method is flexible in its choices of a distance criterion and the numbers of runs and factors, but its performance deteriorates for large problems, as is the case for all computational algorithms. Theoretical construction of maximin distance designs is challenging, and recent efforts are rather technical. Zhou & Xu (2015) examined good lattice point designs with linear transformations, and Wang et al. (2018) further considered nonlinear

Williams transformations of linearly transformed good lattice points. Xiao & Xu (2017) made use of highly specialized objects called Costas arrays.

In this paper we propose a simple method of constructing maximin distance designs. The method is easy to use as it requires only some small maximin distance designs, which can be obtained by either algorithmic search or theoretical construction. Large designs are then constructed using the small designs, and the quality of these large designs, as measured by a maximin distance criterion, is guaranteed by the quality of the small designs. Another appealing feature of our method is that it can be used in conjunction with any distance measure. Application of the method to design construction is demonstrated in a number of scenarios.

2. Notation, definitions and background

A design with N runs, n factors and s levels can be represented by an $N \times n$ matrix with entries from $\{0, 1, \ldots, s-1\}$, and is said to be of U-type if the number of each level appears equally often in every column. We denote such a design by $\mathrm{U}(N, s^n)$. If a $\mathrm{U}(N, s^n)$ has the property that all s^2 ordered pairs of levels occur equally often in any of its $N \times 2$ subarrays, then it is an orthogonal array of strength 2 and will be denoted by $\mathrm{OA}(N, s^n)$. When $N = s$, a $\mathrm{U}(N, s^n)$ becomes an $\mathrm{LH}(N, n)$, a Latin hypercube of N runs for n factors.

Let D be a $\mathrm{U}(N, s^n)$, and let $x_i = (x_{i1}, \ldots, x_{in})$ be the ith row of D. The L_p-distance between rows x_i and x_j is defined to be $d_p(x_i, x_j) = \sum_{k=1}^{n} |x_{ik} - x_{jk}|^p$ for $p \geqslant 1$. Define the L_p-distance of D to be $d_p(D) = \min_{i \neq j} d_p(x_i, x_j)$; that is, $d_p(D)$ is the minimum L_p-distance between any two distinct rows of D. The above definition of L_p-distance, which does not take the pth root as in the standard $(\sum_{k=1}^{n} |x_{ik} - x_{jk}|^p)^{1/p}$ definition, is convenient to use and hence adopted in this work. The maximin distance criteria resulting from our definition and the standard one are equivalent.

The maximin L_p-distance criterion requires us to select a design that maximizes $d_p(D)$ among all competing designs, which are $\mathrm{U}(N, s^n)$ designs in this paper. Zhou & Xu (2015) derived an upper bound on $d_p(D)$ for this class of designs. For any $\mathrm{U}(N, s^n)$, the average L_p-distance between all pairs of points is

$$d_{p,\,\mathrm{ave}} = \frac{n\, C_p}{N^2 - N}, \quad C_p = \sum_{i \neq j} |x_{ik} - x_{jk}|^p. \tag{1}$$

The $d_{p,\,\mathrm{ave}}$ value is constant in the sense that it depends only on N, n and s, but not on any particular design under consideration. We therefore have $d_p(D) \leqslant \lfloor d_{p,\mathrm{ave}} \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer not exceeding x. In particular, we have $d_1(D) \leqslant \lfloor N(s^2 - 1)n/(3Ns - 3s) \rfloor$ for $p = 1$ and $d_2(D) \leqslant \lfloor N(s^2 - 1)n/(6N - 6) \rfloor$ for $p = 2$.

To evaluate a design under the maximin L_p-distance criterion, one can use a distance efficiency defined by $d_p(D)/\lfloor d_{p,\mathrm{ave}} \rfloor$, as recommended in Wang et al. (2018). For general theoretical considerations, however, it is more convenient to use

$$d_{p,\mathrm{eff}}(D) = d_p(D)/d_{p,\mathrm{ave}}, \tag{2}$$

which is the distance efficiency we adopt here. Except for very small designs, there is only a minute difference between these two versions of distance efficiency. When $d_{p,\,\mathrm{eff}}(D) = 1$, the design D is equidistant and is a maximin L_p-distance design. If the design D has a $d_{p,\,\mathrm{eff}}(D)$ value close to 1, it must be a very good design according to the maximin L_p-distance criterion. On the other hand, a maximin L_p-distance design may not have a large $d_{p,\text{eff}}(D)$ value, which occurs when it is impossible to achieve equidistance or near-equidistance.

Our goal is to construct maximin L_p-distance designs. In the process of construction, the Hamming distance will play a supporting role. Consider a $\mathrm{U}(N, s^n)$ design D as introduced above. The Hamming distance $d_{\mathrm{H}}(x_i, x_j)$ between x_i and x_j, the ith and jth rows of D, is the number of components in which they differ. The Hamming distance of design D, denoted by $d_{\mathrm{H}}(D)$, is the minimum Hamming distance between any two distinct rows of D. We can easily find the average of all the Hamming distances between all pairs of rows,

$$d_{\mathrm{H,ave}} = N(s-1)n/\{(N-1)s\}. \tag{3}$$

Similarly, we define

$$d_{\mathrm{H,eff}}(D) = d_{\mathrm{H}}(D)/d_{\mathrm{H,ave}}. \tag{4}$$

3. Method and results

Consider two U-type designs A and B, where A is a $\mathrm{U}(N, s^{n_1})$ and B a $\mathrm{U}(s, q^{n_2})$. From A and B we construct a U-type design D by replacing the uth level of A by the $(u+1)$th row of B for $u = 0, 1, \ldots, s-1$. Then D is a $\mathrm{U}(N, q^{n_1 n_2})$. This method of replacement has its origin in the construction of orthogonal arrays, where it is called an expansive replacement method (Hedayat et al., 1999), and has also been used for different purposes in Mukerjee et al. (2014) and Sun & Tang (2017). It had not, however, previously been considered for the construction of maximin distance designs.

It turns out that very useful results, as shown below, can be obtained regarding the distance properties of design D in relation to those of designs A and B. Although the results are perhaps not surprising, their simplicity and usefulness are unexpected advantages.

Theorem 1. *Suppose that A is a $\mathrm{U}(N, s^{n_1})$ and B is a $\mathrm{U}(s, q^{n_2})$. Let D be a $\mathrm{U}(N, q^{n_1 n_2})$ obtained by replacing the uth level of A by the $(u+1)$th row of B, for $u = 0, 1, \ldots, s-1$. Then:*

(i) $d_p(D) \geqslant d_{\mathrm{H}}(A)\, d_p(B)$;
(ii) $d_{p,\text{eff}}(D) \geqslant d_{\mathrm{H,eff}}(A)\, d_{p,\text{eff}}(B)$.

If B is equidistant under the L_p-distance, then the equalities in (i) *and* (ii) *are both attained.*

The proofs of Theorem 1 and other theoretical results are provided in the Appendix. Although the Hamming distance of design A enters the picture, the distance measure for designs B and D is the L_p-distance for any $p \geqslant 1$. From the proof we see that Theorem 1 actually holds for any additive distance measure, by which we mean a distance of the form $d(x, y) = \sum_{k=1}^{n} d(x_k, y_k)$ for $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$. This feature makes our method very versatile, as one can construct large designs D from small designs B using the L_1-, L_2- or any additive distance.

Theorem 1(i) states that the L_p-distance of D is bounded below by the product of the Hamming distance of A and the L_p-distance of B. If A has a large Hamming distance and B has a large L_p-distance, then design D must also have a large L_p-distance. Theorem 1(ii) says that the distance efficiency of D is bounded below by the product of the distance efficiencies of A and B. If A and B both have distance efficiencies close to 1, then D must also have a distance efficiency close to 1.

Table 1. *Some equidistant maximin designs* $\mathrm{U}(N, s^n)$

A	B	D
$\mathrm{OA}(9, 3^4)$	$\mathrm{LH}(3,3)$	$\mathrm{U}(9,3^{12})$
$\mathrm{OA}(27,3^{13})$	$\mathrm{LH}(3,3)$	$\mathrm{U}(27,3^{39})$
$\mathrm{OA}(81,3^{40})$	$\mathrm{LH}(3,3)$	$\mathrm{U}(81,3^{120})$
$\mathrm{OA}(243,3^{121})$	$\mathrm{LH}(3,3)$	$\mathrm{U}(243,3^{363})$
$\mathrm{OA}(729,3^{364})$	$\mathrm{LH}(3,3)$	$\mathrm{U}(729,3^{1092})$
$\mathrm{OA}(25,5^6)$	$\mathrm{LH}(5,5)$	$\mathrm{U}(25,3^{30})$
$\mathrm{OA}(125,5^{31})$	$\mathrm{LH}(5,5)$	$\mathrm{U}(125,5^{155})$
$\mathrm{OA}(625,5^{156})$	$\mathrm{LH}(5,5)$	$\mathrm{U}(625,5^{780})$
$\mathrm{OA}(64,8^9)$	$\mathrm{LH}(8,8)$	$\mathrm{U}(64,8^{72})$
$\mathrm{OA}(512,8^{73})$	$\mathrm{LH}(8,8)$	$\mathrm{U}(512,8^{584})$
$\mathrm{OA}(81,9^{10})$	$\mathrm{LH}(9,9)$	$\mathrm{U}(81,9^{90})$
$\mathrm{OA}(729,9^{91})$	$\mathrm{LH}(9,9)$	$\mathrm{U}(729,9^{819})$
$\mathrm{OA}(121,11^{12})$	$\mathrm{LH}(11,11)$	$\mathrm{U}(121,11^{132})$
$\mathrm{OA}(1331,11^{133})$	$\mathrm{LH}(11,11)$	$\mathrm{U}(1331,11^{1463})$

The quality of design D is generally better than what is guaranteed by the lower bounds in Theorem 1. The equalities in parts (i) and (ii) of the theorem hold when B is equidistant, but seldom hold otherwise. To see this, observe that the L_p-distance of two rows of design D is the sum of the L_p-distances between a list of pairs of points of design B, with the list corresponding to the different components of the two rows of A that give rise to the two rows of D. To attain the lower bounds in Theorem 1, there must exist two rows of A that, not only have the smallest Hamming distance, as given by $d_{\mathrm{H}}(A)$, but also render, via their different components, a list of pairs of points of B that all have the smallest L_p-distance, as given by $d_p(B)$. These requirements are difficult to meet unless B is equidistant or nearly equidistant.

In Theorem 1, if $d_{\mathrm{H,eff}}(A) = d_{p,\mathrm{eff}}(B) = 1$, then $d_{p,\,\mathrm{eff}}(D) = 1$, in which case all three designs are equidistant.

COROLLARY 1. *If A and B in Theorem 1 are both equidistant, then D is equidistant and hence a maximin distance design.*

If we choose A to be a saturated orthogonal array $\mathrm{OA}(N, s^{n_1})$, then it is equidistant with $d_{\mathrm{H}}(A) = N/s$ according to Cheng (2014, Theorem 8.6). The Rao–Hamming construction gives a saturated $\mathrm{OA}(N, s^{n_1})$ with $N = s^k$ and $n_1 = (N-1)/(s-1)$, and is applicable whenever s is a prime power. Under the L_1-distance, Wang et al. (2018, Theorem 4) showed that an equidistant $\mathrm{LH}(s, s)$, a Latin hypercube of s runs for s factors, can be constructed if $2s + 1$ is a prime. We choose this $\mathrm{LH}(s, s)$ as B. Then we can establish the next theorem.

THEOREM 2. *If s is a prime power and $2s + 1$ is a prime, then an equidistant maximin design $\mathrm{U}(N, s^n)$ where $N = s^k$ and $n = s(s^k - 1)/(s - 1)$ can be constructed under the L_1-distance for every integer $k \geqslant 2$.*

Some designs obtained by applying Theorem 2 are presented in Table 1.

The equidistant maximin designs in Theorem 2 and Table 1 are supersaturated in that $n = s^k + s^{k-1} + \cdots + s > N = s^k$. The rest of the section is devoted to the construction of designs that are not supersaturated. We will examine three other choices of B, while still using the saturated $\mathrm{OA}(s^k, s^{n_1})$ with $n_1 = (s^k - 1)/(s - 1)$ as A.

From the $\mathrm{LH}(s, s)$ above, Wang et al. (2018) constructed an $\mathrm{LH}(s + 1, s)$ with $d_{1,\mathrm{eff}} = (s + 1)/(s + 2)$. Replacing s by $s - 1$, we obtain an $\mathrm{LH}(s, s - 1)$ with $d_{1,\mathrm{eff}} = s/(s + 1)$. Using this Latin hypercube as B, we obtain the next result.

Table 2. *Some* $\mathrm{U}(N, s^n)$*designs with high* L_1*-distance efficiencies*

A	B	D	$d_{1,\mathrm{eff}}(D)$	Source
$\mathrm{OA}(16,4^5)$	$\mathrm{LH}(4,3)$	$\mathrm{U}(16,4^{15})$	0.800	Proposition 1
$\mathrm{OA}(64,4^{21})$	$\mathrm{LH}(4,3)$	$\mathrm{U}(64,4^{63})$	0.800	Proposition 1
$\mathrm{OA}(256,4^{85})$	$\mathrm{LH}(4,3)$	$\mathrm{U}(256,4^{255})$	0.800	Proposition 1
$\mathrm{OA}(25,5^6)$	$\mathrm{LH}(5,4)$	$\mathrm{U}(25,5^{24})$	0.875	Proposition 2
$\mathrm{OA}(125,5^{31})$	$\mathrm{LH}(5,4)$	$\mathrm{U}(125,5^{124})$	0.875	Proposition 2
$\mathrm{OA}(625,5^{156})$	$\mathrm{LH}(5,4)$	$\mathrm{U}(625,5^{624})$	0.875	Proposition 2
$\mathrm{OA}(49,7^8)$	$\mathrm{LH}(7,6)$	$\mathrm{U}(49,7^{48})$	1	Proposition 2
$\mathrm{OA}(343,7^{57})$	$\mathrm{LH}(7,6)$	$\mathrm{U}(343,7^{342})$	1	Proposition 2
$\mathrm{OA}(81,9^{10})$	$\mathrm{LH}(9,8)$	$\mathrm{U}(81,9^{80})$	0.900	Proposition 1
$\mathrm{OA}(729,9^{91})$	$\mathrm{LH}(9,8)$	$\mathrm{U}(729,9^{728})$	0.900	Proposition 1
$\mathrm{OA}(121,11^{12})$	$\mathrm{LH}(11,10)$	$\mathrm{U}(121,11^{120})$	0.975	Proposition 2
$\mathrm{OA}(169,13^{14})$	$\mathrm{LH}(13,12)$	$\mathrm{U}(169,13^{168})$	0.929	Proposition 2
$\mathrm{OA}(256,16^{17})$	$\mathrm{LH}(16,15)$	$\mathrm{U}(256,16^{255})$	0.941	Proposition 1
$\mathrm{OA}(289,17^{18})$	$\mathrm{LH}(17,16)$	$\mathrm{U}(289,17^{288})$	0.979	Proposition 2
$\mathrm{OA}(361,19^{20})$	$\mathrm{LH}(19,18)$	$\mathrm{U}(361,19^{360})$	0.958	Proposition 2

PROPOSITION 1. *If s is a prime power and $2s-1$ is a prime, then a* $\mathrm{U}(N, s^n)$ *design D, where $N = s^k$ and $n = s^k - 1$, can be constructed to have $d_{1,\mathrm{eff}}(D) \geqslant s/(s+1)$ for every integer $k \geqslant 2$.*

Proposition 1 provides a class of designs with high distance efficiency. We see that $d_{1,\mathrm{eff}}(D) \geqslant 0.9$ for $s \geqslant 9$ and that $d_{1,\mathrm{eff}}(D) \geqslant 0.8$ even for $s = 4$. Wang et al. (2018, Theorem 2) constructed another $\mathrm{LH}(s, s-1)$ when s is prime. If we take this design as our choice of B, then it has $d_{1,\mathrm{eff}}(B) \geqslant 1 - 2/(3s^2 - 3)^{1/2}$ according to Wang et al. (2018, Theorem 2), leading to another construction of designs with high distance efficiency.

PROPOSITION 2. *If s is a prime, then a* $\mathrm{U}(N, s^n)$ *design D where $N = s^k$ and $n = s^k - 1$ can be constructed to have $d_{1,\mathrm{eff}}(D) \geqslant 1 - 2/(3s^2 - 3)^{1/2}$ for every integer $k \geqslant 2$.*

Propositions 1 and 2 together cover many s values for a $\mathrm{U}(N, s^n)$ with high distance efficiency to be constructed. We give some examples. For $s = 5$, Proposition 2 is applicable, but Proposition 1 is not. In contrast, for $s = 9$, Proposition 1 is applicable, but Proposition 2 is not. For $s = 7$, both Propositions 1 and 2 are applicable. Proposition 1 gives a design with $d_{1,\mathrm{eff}}(D) \geqslant 0.875$ while Proposition 2 gives a design with $d_{1,\mathrm{eff}}(D) \geqslant 0.833$. Upon examining the two $\mathrm{LH}(7, 6)$ designs used in Propositions 1 and 2, we were pleasantly surprised to find that the $\mathrm{LH}(7, 6)$ constructed by Theorem 2 of Wang et al. (2018) is equidistant. This means that the design from Proposition 2 for $s = 7$ actually has $d_{1,\mathrm{eff}}(D) = 1$. The equidistant $\mathrm{LH}(7, 6)$ is displayed below:

$$\begin{array}{cccccc}
3 & 1 & 0 & 2 & 4 & 6 \\
1 & 2 & 6 & 3 & 0 & 4 \\
0 & 6 & 1 & 4 & 3 & 2 \\
2 & 3 & 4 & 1 & 6 & 0 \\
4 & 0 & 3 & 6 & 2 & 1 \\
6 & 4 & 2 & 0 & 1 & 3 \\
5 & 5 & 5 & 5 & 5 & 5
\end{array}$$

Table 2 presents some designs obtained by Propositions 1 and 2.

We conclude this section with a result under the L_2-distance. Zhou et al. (2020) constructed an $\mathrm{LH}(2^k, 2^{k-1})$ with L_2-distance efficiency $d_{2,\mathrm{eff}} \geqslant 1 - 1/2^k$ for any $k \geqslant 2$. We now use this design as B while still taking a saturated orthogonal array as A.

PROPOSITION 3. *A* $\mathrm{U}(N, s^n)$ *design* D *where* $s = 2^{k_2}$, $N = 2^{k_1 k_2}$ *and* $n = 2^{k_2-1}(2^{k_1 k_2} - 1)/(2^{k_2} - 1)$ *can be constructed to have* $d_{2,\mathrm{eff}}(D) \geqslant 1 - 1/2^{k_2}$ *for all* $k_1 \geqslant 2$ *and* $k_2 \geqslant 2$.

Taking $k_1 = 2$ and $k_2 = 3$ gives a $\mathrm{U}(64, 8^{36})$ with $d_{2,\mathrm{eff}}(D) \geqslant 7/8$. If we take $k_1 = 2$ and $k_2 = 4$, we obtain a $\mathrm{U}(256, 16^{136})$ with $d_{2,\mathrm{eff}}(D) \geqslant 15/16$.

4. FURTHER RESULTS

In the previous section following the main result, Theorem 1, we focused on the construction of design D under the condition that A is a saturated orthogonal array, which is equidistant under the Hamming distance. We shift gears in this section and consider the construction of a design D with B chosen to be equidistant under the L_p-distance. Our candidate choices for A here are the subarrays of a saturated orthogonal array.

Let S denote a saturated orthogonal array $\mathrm{OA}(N, s^{n_1})$ where $n_1 = (N-1)/(s-1)$, given by the Rao–Hamming construction, which is available whenever s is a prime power. Let design A be obtained by selecting some columns from S. Those columns not selected by A form a complementary design of A, which we denote by $\bar{A} = S \setminus A$. Let $\bar{A}$ and A have j and $n_1 - j$ columns, respectively. As S is equidistant with $d_{\mathrm{H}}(S) = N/s$, we have that $d_{\mathrm{H}}(A) = N/s - d^*_{\mathrm{H}}(\bar{A})$, where $d^*_{\mathrm{H}}(\bar{A})$ is the maximum Hamming distance between any two rows of $\bar{A}$.

LEMMA 1. *Let* B *be an equidistant* $\mathrm{U}(s, q^{n_2})$ *according to the* L_p*-distance for any* $p \geqslant 1$*, and let* A *be a subarray of* S *as given above. Then design* D *in Theorem* 1 *is a* $\mathrm{U}(N, q^{(n_1-j)n_2})$ *with* L_p*-distance efficiency*

$$d_{p,\mathrm{eff}}(D) = d_{\mathrm{H,eff}}(A) = \left(\frac{N-1}{N}\right)\left(\frac{N - s\, d^*_{\mathrm{H}}(\bar{A})}{N - js + j - 1}\right),$$

where j *is the number of columns of* $\bar{A}$ *and* $n_1 = (N-1)/(s-1)$.

Obviously, for $j = 1, 2$ we have $d^*_{\mathrm{H}}(\bar{A}) = j$ for any choice of $\bar{A}$ with j columns. For $j \geqslant 3$ we have $d^*_{\mathrm{H}}(\bar{A}) \leqslant j$. We therefore have that for $j \geqslant 1$,

$$d_{p,\mathrm{eff}}(D) = d_{\mathrm{H,eff}}(A) \geqslant \left(\frac{N-1}{N}\right)\left(\frac{N - js}{N - js + j - 1}\right),$$

which is close to 1 provided that j is small.

For given j, we want to find an $\bar{A}$ that minimizes $d^*_{\mathrm{H}}(\bar{A})$, thus maximizing $d_{p,\mathrm{eff}}(D) = d_{\mathrm{H,eff}}(A)$ owing to Lemma 1. We have seen already that any $\bar{A}$ minimizes $d^*_{\mathrm{H}}(\bar{A})$ if $j = 1, 2$. Although it may be too difficult to solve this optimization problem for all j, it is possible to find a solution for certain j values. We discuss this next.

The saturated orthogonal array S, the $\mathrm{OA}(N, s^{n_1})$ with $N = s^k$ and $n_1 = (N-1)/(s-1)$ introduced above, is generated by k independent columns $e_1, \ldots, e_k$, and it collects these k independent columns and all their possible interaction columns. The columns of S can be represented by $e_1^{u_1} \cdots e_k^{u_k}$, with the first nonzero entry u_j in the vector $(u_1, \ldots, u_k)$ equal to 1. For any $k_0 = 1, \ldots, k-1$, let S_0 be obtained by collecting all those columns in S that correspond to $u_{k_0+1} = \cdots = u_k = 0$. Effectively, S_0 is generated by k_0 independent columns $e_1, \ldots, e_{k_0}$. We see that S_0 is a saturated orthogonal array $\mathrm{OA}(N_0, s^{n_0})$ where $N_0 = s^{k_0}$ and $n_0 = (s^{k_0} - 1)/(s-1)$, with its runs all replicated s^{k-k_0} times. This shows that $d_{\mathrm{H}}^*(S_0) = N_0/s = s^{k_0-1}$. Taking $\bar{A} = S_0$ gives the main result of this section.

THEOREM 3. *Let B be an equidistant $\mathrm{U}(s, q^{n_2})$ under the L_p-distance. Take $\bar{A} = S_0$ so that $A = S \setminus S_0$. Then design D in Theorem 1 is a $\mathrm{U}(N, q^{(n_1-n_0)n_2})$ that has*

$$d_{p,\mathrm{eff}}(D) = d_{\mathrm{H,eff}}(A) = 1 - 1/N,$$

where $n_1 - n_0 = (s^k - s^{k_0})/(s-1)$. This choice of $\bar{A}$ is optimal in that no other $\bar{A}$ can give a higher value of $d_{\mathrm{H,eff}}(A)$.

We now take B to be the $\mathrm{LH}(s, s)$ from Wang et al. (2018), which is equidistant under the L_1-distance.

COROLLARY 2. *Let s be a prime power such that $2s + 1$ is prime. If we take B to be the $\mathrm{LH}(s, s)$ from Wang et al. (2018) and A to be $S \setminus S_0$ as in Theorem 3, then we obtain a $\mathrm{U}(N, s^n)$ with $n = s(s^k - s^{k_0})/(s-1)$ that has $d_{1,\mathrm{eff}}(D) = 1 - 1/N$. Of special interest is the case where $k_0 = k - 1$, for which we obtain a $\mathrm{U}(N, s^N)$ with $d_{1,\mathrm{eff}}(D) = 1 - 1/N$.*

Example 1. Take $s = 3$ and $k = 3$ in Corollary 2. Then design S is a saturated $\mathrm{OA}(27, 3^{13})$ whose set of 13 columns is

$$S = \{e_1, e_2, e_1e_2, e_1e_2^2, e_3, e_1e_3, e_1e_3^2, e_2e_3, e_2e_3^2, e_1e_2e_3, e_1e_2^2e_3, e_1e_2e_3^2, e_1e_2^2e_3^2\}.$$

If we take $k_0 = 2$, then $\bar{A} = S_0 = \{e_1, e_2, e_1e_2, e_1e_2^2\}$; so $A = S \setminus \bar{A}$ is an $\mathrm{OA}(27, 3^9)$. Using this A in conjunction with an equidistant $\mathrm{LH}(3, 3)$ as B, we obtain design D, a $\mathrm{U}(27, 3^{27})$ with $d_{1,\mathrm{eff}}(D) = 26/27 = 0.963$. Design D is actually a maximin L_1-distance design because we can calculate that $d_1(D) = 24$ and $d_{1,\mathrm{ave}}(D) = 324/13 < 25$, showing that the L_1-distance of D attains the upper bound $\lfloor 324/13 \rfloor = 24$.

Table 3 provides a list of $\mathrm{U}(N, s^N)$ designs that can be constructed by Corollary 2. These $\mathrm{U}(N, s^N)$ are supersaturated, so deleting one column gives a $\mathrm{U}(N, s^{N-1})$. The last column of Table 3 gives information on the distance efficiency for these $\mathrm{U}(N, s^{N-1})$.

Theorem 3 provides an optimal choice of $\bar{A}$ for a set of selected j values, $j = (s^{k_0} - 1)/(s-1)$ for $k_0 = 1, \ldots, k-1$, where j denotes the number of columns in $\bar{A}$. For any other j value, a sensible strategy of connecting the dots, as suggested by the same theorem, is to construct $\bar{A}$ by selecting the columns of S in Yates order. We use an example to illustrate this idea.

Example 2. For $s = 5$ and $k = 3$, S is a saturated $\mathrm{OA}(125, 5^{31})$ whose 31 columns are given in Yates order in Table 4, where column $e_1^{u_1} \cdots e_k^{u_k}$ is replaced by $1^{u_1} \cdots k^{u_k}$ for simplicity. We sequentially delete the first j columns in S according to Yates order and take the remaining $n = 31 - j$ columns as A. Using this A together with the equidistant $\mathrm{LH}(5, 5)$, we obtain a $\mathrm{U}(125, 5^{5n})$ for $n = 31, 30, 29, \ldots$. Table 5 gives the L_1-distance efficiencies of these designs for $n = 31, 30, \ldots, 19$.

Table 3. *The L_1-distance efficiencies of an $N \times N$ design D in Corollary 2 and an $N \times (N-1)$ design D_{-1} upon deleting any one column of D*

A	B	D	$d_{1,\text{eff}}(D)$	$d_{1,\text{eff}}(D_{-1})$
$\text{OA}(9,3^3)$	$\text{LH}(3,3)$	$\text{U}(9,3^9)$	0.889	0.750
$\text{OA}(27,3^9)$	$\text{LH}(3,3)$	$\text{U}(27,3^{27})$	0.963	0.917
$\text{OA}(81,3^{27})$	$\text{LH}(3,3)$	$\text{U}(81,3^{81})$	0.988	0.972
$\text{OA}(243,3^{81})$	$\text{LH}(3,3)$	$\text{U}(243,3^{243})$	0.996	0.991
$\text{OA}(729,3^{243})$	$\text{LH}(3,3)$	$\text{U}(729,3^{729})$	0.999	0.997
$\text{OA}(25,5^5)$	$\text{LH}(5,5)$	$\text{U}(25,5^{25})$	0.960	0.900
$\text{OA}(125,5^{25})$	$\text{LH}(5,5)$	$\text{U}(125,5^{125})$	0.992	0.980
$\text{OA}(625,5^{125})$	$\text{LH}(5,5)$	$\text{U}(625,5^{625})$	0.998	0.996
$\text{OA}(64,8^8)$	$\text{LH}(8,8)$	$\text{U}(64,8^{64})$	0.984	0.958
$\text{OA}(512,8^{64})$	$\text{LH}(8,8)$	$\text{U}(512,8^{512})$	0.998	0.995
$\text{OA}(81,9^9)$	$\text{LH}(9,9)$	$\text{U}(81,9^{81})$	0.988	0.967
$\text{OA}(729,9^{81})$	$\text{LH}(9,9)$	$\text{U}(729,9^{729})$	0.999	0.996
$\text{OA}(121,11^{11})$	$\text{LH}(11,11)$	$\text{U}(121,11^{121})$	0.992	0.977
$\text{OA}(1331,11^{121})$	$\text{LH}(11,11)$	$\text{U}(1331,11^{1331})$	0.999	0.998

Table 4. *The* 31 *columns of* $\text{OA}(125, 5^{31})$

1	2	12	12^2	12^3	12^4	3	13	13^2	13^3	13^4	23	23^2	23^3	23^4	123
12^23	12^33	12^43	123^2	12^23^2	12^33^2	12^43^2	123^3	12^23^3	12^33^3	12^43^3	123^4	12^23^4	12^33^4	12^43^4	

Table 5. *The L_1-distance efficiencies of* $\text{U}(125, 5^{5n})$ *for* $n = 31, 30, \ldots, 19$

$5n$	155	150	145	140	135	130	125	120	115	110	105	100	95
$d_{1,\text{eff}}(D)$	1	0.992	0.983	0.974	0.964	0.954	0.992	0.982	0.970	0.958	0.945	0.930	0.914

In the remainder of this section, we compare our designs with existing designs in the literature. We consider good lattice point designs and their linear transformations (Zhou & Xu, 2015), designs obtained by further Williams transformations (Wang et al., 2018), and designs from the `R` (R Development Core Team, 2021) packages `SLHD` (Ba, 2013) and `lhs` (Carnell, 2020). All designs are scaled into $[0,1]^n$ and the L_1-distance is used for comparison. The results, given in Table 6, show that our designs are uniformly better, which is somewhat surprising even to us.

Our designs are obtained in the same manner as in Example 2, where A is obtained by deleting columns in Yates order from a saturated orthogonal array and B is taken to be an L_1-equidistant $\text{U}(s, s^{n_2})$. To use the `R` package `SLHD` (Ba, 2013), we run the `R` command `maximinSLHD` 100 times and then select the best design out of the 100 generated designs. For $N \leqslant 125$ the default setting of 10^6 iterations is used, but for $N > 125$ the algorithm is considerably slower so we use 10^4 iterations to save time. For the `R` package `lhs` (Carnell, 2020), we run the `R` command `maximinLHS` 100 times with default settings and choose the best sample under the L_1-distance.

If B is equidistant, then (B, B) is also equidistant. Therefore, designs with larger n for a given N are readily available. Table 7 compares our designs with those constructed by the `R` packages `SLHD` and `lhs` in the range $N/5 \leqslant n < N$. Selected n values are such that the ratio N/n is approximately equal to 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5. We see that `SLHD` is always better than `lhs`. Our method underperforms `SLHD` for small n values, but becomes dominant as soon as the ratio N/n exceeds a certain threshold. This phenomenon also occurs for designs of other N values that are not reported here to save space.

Table 6. *L_1-distances of various $N \times n$ designs*

N	s	n	Ours	GLP	LGLP	WGLP	SLHD	lhs
25	5	20	7.5	5.3	6.1	6.8	6.3	5.1
49	7	42	16.0	10.8	12.3	14.3	13.0	10.7
64	8	32	10.3	8.1	8.1	9.3	9.3	7.1
81	9	54	18.8	13.7	15.2	17.7	16.3	13.0
121	11	110	39.6	27.8	33.2	36.9	34.4	27.8
125	5	100	37.5	25.2	29.4	32.7	30.8	24.7
343	7	294	112.0	73.7	84.3	97.8	85.8	76.5
512	8	256	85.7	64.1	64.1	74.4	72.6	63.2
729	9	486	172.5	121.7	135.2	157.5	142.4	127.2

Ours, our designs; GLP, good lattice point designs; LGLP, linear transformation of GLP; WGLP, Williams transformation of LGLP; SLHD, `R` package `SLHD`; lhs, `R` package `lhs`.

Table 7. *L_1-distances of $N \times n$ designs with varying N/n ratios*

N	s	n	25	25	30	35	40	50	60	80	124
125	5	Ours	2.5	2.5	5.0	7.5	10.0	15.0	20.0	30.0	49.0
		SLHD	6.2	6.2	7.7	9.3	11.2	14.5	17.7	24.1	39.1
		lhs	4.5	4.5	5.7	7.3	8.2	11.1	13.7	19.1	32.0
		n	66	72	84	96	114	132	168	228	342
343	7	Ours	13.3	16.0	21.3	26.7	34.7	42.7	58.7	82.7	131.0
		SLHD	16.1	17.7	21.1	24.6	31.1	36.4	47.4	65.5	101.0
		lhs	12.8	14.2	17.2	20.6	25.3	30.1	40.4	57.5	91.2

He (2019) introduced interleaved lattice-based maximin L_2-distance designs and provided an `R` package `InterleavedMaximinD` for implementation of his method. Table 8 compares our designs obtained from Proposition 3 with the designs generated by `InterleavedMaximinD`, for $N = 256$ and $n = 136, 170$ under both L_2- and L_1-distances. For reference, corresponding designs from the `R` package `SLHD` are also included. We see that the designs from `InterleavedMaximinD` are the best under the L_2-distance while our designs are the best under the L_1-distance. The superior performance of `InterleavedMaximinD` under the L_2-distance appears to be due to the corresponding designs having only two levels, which may be an undesirable feature for computer experiments. We have also attempted, but found it infeasible, to compare our designs with interleaved lattice-based maximin distance designs for other run sizes, as `InterleavedMaximinD` very frequently does not generate designs with the required run sizes. For example, `InterleavedMaximinD` outputs a design with 128 runs when one inputs $N = 64$ and $n = 36$, and a design with 567 runs when one inputs $N = 512$ and $n = 192$.

In recent work, He (2020) examined another attractive class of lattice-based designs that aim for a large separation distance on all projections. These designs are most suitable when there are relatively small numbers of factors compared to run sizes. He (2020) considered $n = 4, 8, 16$, but the method became difficult and slow to implement for $n = 16$ factors.

5. Concluding remarks

The central theme of this article is the construction of designs with large distances. Besides having large distances, our constructed designs also enjoy some other space-filling properties. It is obvious that, given an orthogonal array A, if B has orthogonal columns, then design D must

Table 8. *Comparisons with designs obtained from* `InterleavedMaximinD` *in terms of both the L_1-distance d_1 and the L_2-distance d_2*

N	n	Design	s	d_2	d_1
256	136	SLHD	256	17.5	37.9
		Ours	16	24.2	38.4
		Lattice	2	30	30
256	170	SLHD	256	22.6	48.4
		Ours	4	35.6	64
		Lattice	2	37	37

SLHD, `R` package `SLHD`; Ours, our designs; Lattice, `R` package `InterleavedMaximinD`.

also have orthogonal columns. Moreover, our designs have a property that resembles mappable nearly orthogonal arrays (Mukerjee et al., 2014), in that any two columns of D that come from two different columns of A form an orthogonal array, implying that the design D is space-filling in most of two dimensions.

There are a number of further issues that would be interesting to explore. The present paper has focused on designs constructed when at least one of A and B is equidistant. Theorem 1 guarantees that the resulting design has a large distance efficiency when A and B both have large distance efficiencies, but detailed and in-depth studies could potentially shed more light on the construction of maximin distance designs. Unless B is equidistant, level permutation of A has an effect on the distance of D, a point alluded to, somewhat implicitly, in the discussion after Theorem 1, and this effect becomes more pronounced if B is far from being equidistant. It would be very interesting to investigate this important scenario in the future.

Acknowledgement

Tang was supported by the Natural Sciences and Engineering Research Council of Canada. Li and Liu were supported by the National Natural Science Foundation of China, the National Ten Thousand Talents Program of China, the Tianjin Development Program for Innovation and Entrepreneurship, and the Tianjin '131' Talents programme. The authors are listed in alphabetical order.

Appendix

Proof of Theorem 1

Let c_i and c_j be two rows of D where $i \neq j$, and let a_i and a_j be the corresponding two rows of A. A moment of thought gives that $d_p(c_i, c_j) \geqslant d_{\mathrm{H}}(a_i, a_j) d_p(B) \geqslant d_{\mathrm{H}}(A) d_p(B)$. This shows that $d_p(D) \geqslant d_{\mathrm{H}}(A) d_p(B)$. Thus part (i) of Theorem 1 is established.

To prove part (ii), recall from § 2 the average distance for a $\mathrm{U}(N, q^n)$ given in (1). Since

$$C_p = \sum_{i \neq j} |x_{ik} - x_{jk}|^p = 2N^2\{1^p \times (q-1) + 2^p \times (q-2) + \cdots + (q-1)^p \times 1\}/q^2,$$

we have $d_{p,\mathrm{ave}} = nNW(q)/(N-1)$ where $W(q) = 2\sum_{i=1}^{q-1} i^p(q-i)/q^2$. Applying this to designs B and D, we obtain $d_{p,\mathrm{ave}}(B) = n_2 sW(q)/(s-1)$ and $d_{p,\mathrm{ave}}(D) = (n_1 n_2)NW(q)/(N-1)$. For design A, we have from (3) in § 2 that $d_{\mathrm{H,ave}}(A) = N(s-1)n_1/\{(N-1)s\}$. Therefore $d_{p,\mathrm{ave}}(D) = d_{\mathrm{H,ave}}(A) d_{p,\mathrm{ave}}(B)$. This implies that $d_{p,\mathrm{eff}}(D) = d_p(D)/d_{p,\mathrm{ave}}(D) \geqslant d_{\mathrm{H,eff}}(A) d_{p,\mathrm{eff}}(B)$ by part (i) and the definitions of $d_{\mathrm{H,eff}}(A)$ in (4) and $d_{p,\mathrm{eff}}(B)$ in (2). The last statement in the theorem is obvious from the proof of part (i).

Proof of Lemma 1

Since B is equidistant under the L_p-distance, we have that $d_{p,\text{eff}}(B) = 1$. Therefore $d_{p,\text{eff}}(D) = d_{\text{H,eff}}(A)$. We also know that $d_{\text{H}}(A) = N/s - d^*_{\text{H}}(\bar{A})$.

As A is a $\text{U}(N, s^{(n_1-j)})$ where $n_1 = (N-1)/(s-1)$ and j is the number of columns in $\bar{A}$, again by (3) we have $d_{\text{H,ave}}(A) = N(s-1)(n_1-j)/\{(N-1)s\}$. Therefore

$$d_{p,\text{eff}}(D) = d_{\text{H,eff}}(A) = d_{\text{H}}(A)/d_{\text{H,ave}}(A) = \left(\frac{N-1}{N}\right)\left(\frac{N - s\, d^*_{\text{H}}(\bar{A})}{N - js + j - 1}\right),$$

where $n_1(s-1) = N-1$ has helped with the simplification in the last step.

Proof of Theorem 3

It is easy to see that $d_{\text{H,eff}}(A) = (N-1)/N$, because $d_{\text{H}}(A) = N/s - N_0/s$ and $d_{\text{H,ave}}(A) = N(s-1)\{(N-1)/(s-1) - (N_0-1)/(s-1)\}/\{(N-1)s\} = N(N-N_0)/\{(N-1)s\}$. No better $\bar{A}$ can be found because $d_{\text{H,ave}}(A) - d_{\text{H}}(A) = (N-N_0)/\{(N-1)s\} < 1$.

References

Ba, S. (2013). *SlhD: Maximin-Distance (Sliced) Latin Hypercube Designs*. R package version 1.2-1, available at http://cran.r-project.org/package=SlhD.

Carnell, R. (2020). *lhs: Latin Hypercube Samples*, R package version 1.0.2, available at https://CRAN.R-project.org/package=lhs.

Cheng, C. S. (2014). *Theory of Factorial Design: Single- and Multi-Stratum Experiments*. Boca Raton, Florida: CRC Press.

Fang, K. T., Li, R. & Sudjianto, A. (2006). *Design and Modeling for Computer Experiments*. Boca Raton, Florida: Chapman & Hall/CRC.

He, X. (2019). Interleaved lattice-based maximin distance designs. *Biometrika* **106**, 453–64.

He, X. (2020). Lattice-based designs with quasi-optimal separation distance on all projections. *Biometrika* **108**, 443–45.

He, Y., Cheng, C. S. & Tang, B. (2018). Strong orthogonal arrays of strength two plus. *Ann. Statist.* **46**, 457–68.

Hedayat, A. S., Sloane, N. J. A. & Stufken, J. (1999). *Orthogonal Arrays: Theory and Applications*. New York: Springer.

Johnson, M. E., Moore, L. M. & Ylvisaker, D. (1990). Minimax and maximin distance designs. *J. Statist. Plan. Infer.* **26**, 131–48.

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**, 239–45.

Mukerjee, R., Sun, F. & Tang, B. (2014). Nearly orthogonal arrays mappable into fully orthogonal arrays. *Biometrika* **101**, 957–63.

Owen, A. B. (1992). Orthogonal arrays for computer experiments, integration and visualization. *Statist. Sinica* **2**, 439–52.

R Development Core Team (2021). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0, http://www.R-project.org.

Santner, T. J., Williams, B. J. & Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. New York: Springer.

Sun, F. & Tang, B. (2017). A method of constructing space-filling orthogonal designs. *J. Am. Statist. Assoc.* **112**, 683–9.

Tang, B. (1993). Orthogonal array-based Latin hypercubes. *J. Am. Statist. Assoc.* **88**, 1392–7.

Wang, L., Xiao, Q. & Xu, H. (2018). Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs. *Ann. Statist.* **46**, 3741–66.

Xiao, Q. & Xu, H. (2017). Construction of maximin distance Latin squares and related Latin hypercube designs. *Biometrika* **104**, 455–64.

Zhou, W., Yang, J. F. & Liu, M. Q. (2020). Optimal maximin L_2-distance Latin hypercube designs. *J. Statist. Plan. Infer.* **207**, 113–22.

Zhou, Y. & Xu, H. (2015). Space-filling properties of good lattice point sets. *Biometrika* **102**, 959–66.

[*Received on* 28 *May* 2020. *Editorial decision on* 12 *October* 2020]