A method of constructing maximin distance designs

BY WENLONG LI, MIN-QIAN LIU
School of Statistics and Data Science, LPMC & KLMDASR, Nankai University,
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

One attractive class of space-filling designs for computer experiments is that of maximin distance designs. Algorithmic search for such designs is commonly used but this method becomes ineffective for large problems. Theoretical construction of maximin distance designs is challenging; some results have been obtained recently, often by employing highly specialized techniques. This paper presents an easy-to-use method for constructing maximin distance designs. The method is versatile as it is applicable for any distance measure. Our 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 to investigate the complex systems in engineering and sciences. Most commonly used designs for computer experiments are space-filling designs (Fang et al., 2006; Santner et al., 2003), which aim to scatter the design points over the design space as uniformly as possible. We may construct designs that are space-filling in low dimensions by using orthogonal arrays or stronger versions of such arrays. Research in this area started from McKay et al. (1979), continued in Owen (1992) and Tang (1993), and remains exuberant to date. For some recent developments, we refer to Mukerjee et al. (2014) and He et al. (2018).

Maximin distance designs, first introduced in Johnson et al. (1990), are also popular choices for designing computer experiments. Johnson et al. (1990) showed that maximin distance designs are asymptotically optimal under 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 that of Ba (2013). An algorithmic search method is flexible in the choices of a distance criterion as well as 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 work is seen to be 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.
The purpose of this paper is to present a simple method of constructing maximin distance designs. The method is easy to use as we only need some small maximin distance designs, which can be obtained by either algorithmic search or theoretical construction. Large designs are then constructed using small designs, the quality of large designs, as measured by a maximin distance criterion, is guaranteed by the quality of the small designs. Another salient feature of our method is that it can be used in conjunction with any measure of distance. Application of the method to design construction is considered under 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 U-type if the number of every level appears equally often in every column. We use \( U(N, s^n) \) to denote such a design. If a \( U(N, s^n) \) 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 \( OA(N, s^n) \) in this paper. When \( N = s \), a \( U(N, s^n) \) becomes an LH(N, n), a Latin hypercube of \( N \) runs for \( n \) factors.

Let \( D \) be a \( U(N, s^n) \) and let \( x_{ij} = (x_{i1}, \ldots, x_{in}) \) be the \( i \)th 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 \( p \geq 1 \). 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 \( p \)th root as in the standard \( \left( \sum_{k=1}^{n} |x_{ik} - x_{jk}|^p \right)^{1/p} \), is convenient to use and thus adopted. The resulting maximin distance criteria 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 \( U(N, s^n) \)'s in this paper. Zhou & Xu (2015) derived an upper bound for \( d_p(D) \) among this class of designs. For any \( U(N, s^n) \), its average \( L_p \)-distance among all pairs of points is
\[
d_{p,\text{ave}} = nC_p/(N^2 - N)
\]
with
\[
C_p = \sum_{i \neq j} |x_{ik} - x_{jk}|^p.
\]
The \( d_{p,\text{ave}} \) value is a constant in that it only depends on \( N, n, s \) but not on any particular design under consideration. We therefore have \( d_p(D) \leq |d_{p,\text{ave}}| \) where \( \lceil x \rceil \) is the largest integer not exceeding \( x \). In particularly, we have
\[
d_1(D) \leq \lceil N(s^2 - 1)n/(3Ns - 3s) \rceil \quad \text{for} \quad p = 1 \quad \text{and}
\]
\[
d_2(D) \leq \lceil N(s^2 - 1)n/(6N - 6) \rceil \quad \text{for} \quad p = 2.
\]

To evaluate a design under the maximin \( L_p \)-distance criterion, we may use a distance efficiency given by \( d_p(D)/|d_{p,\text{ave}}| \), as recommended in Wang et al. (2018). In general theoretical discussion, however, it is more convenient to use
\[
d_{p,\text{eff}}(D) = d_p(D)/d_{p,\text{ave}}
\]
as is the case in this paper. Except for very small designs, there is only a minute difference between the two versions of distance efficiency. When \( d_{p,\text{eff}}(D) = 1 \), design \( D \) is equi-distant and a maximin \( L_p \)-distance design. If design \( D \) has a \( d_{p,\text{eff}}(D) \) 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 happens when it is impossible to achieve equi-distance or near equi-distance.

Our purpose is to construct maximin \( L_p \)-distance designs. In the process of construction, the Hamming distance will play a supporting role. Still consider a design \( D \), a \( U(N, s^n) \) as introduced above. The Hamming distance \( d_H(x_i, x_j) \) between \( x_i \) and \( x_j \), the \( i \)th and \( j \)th rows of \( D \) is the number of components where they differ. The Hamming distance of design \( D \), denoted by
Maximin distance designs

\[ d_H(D) \], is the minimum Hamming distance between any two distinct rows of \( D \). We can easily find the average of all the Hamming distances among all pairs of rows, which is given by

\[ d_{H,\text{ave}} = N(s - 1)n/((N - 1)s). \]

Similarly, we define

\[ d_{H,\text{eff}}(D) = d_H(D)/d_{H,\text{ave}}. \]

3. Method and results

Consider two U-type designs \( A \) and \( B \), where \( A \) is a \( U(N, s^{n_1}) \) and \( B \) a \( U(s, q^{n_2}) \). From \( A \) and \( B \), we construct a U-type design \( D \) by replacing the \( u \)th level of \( A \) by the \((u + 1)\)th row of \( B \) for \( u = 0, 1, \ldots, s - 1 \). Then \( D \) is a \( U(N, q^{n_1}q^{n_2}) \). This method of replacement has an 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), but has not previously been considered for the construction of maximin distance designs.

It turns out that very useful results, as given below, can be obtained regarding the distance properties of design \( D \) in relation with those of \( A \) and \( B \). Although the results are perhaps not surprising, their simplicity and usefulness are something one can hardly foresee.

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

(a) \( d_p(D) \geq d_H(A) d_p(B) \), and

(b) \( d_{p,\text{eff}}(D) \geq d_{H,\text{eff}}(A) d_{p,\text{eff}}(B) \).

If \( B \) is equi-distant under the \( L_p \)-distance, then the equalities in (a) and (b) are both attained.

The proofs of Theorem 1 and later results are provided in an appendix. While the Hamming distance of design \( A \) enters the picture, the distance measure for designs \( B \) and \( D \) is the \( L_p \)-distance for any \( p \geq 1 \). From the proof, we actually see that Theorem 1 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 salient feature of Theorem 1 makes it versatile as one can construct large design \( D \) from small design \( B \) using the \( L_1 \), \( L_2 \) or any additive distance.

Part (a) of Theorem 1 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. Part (b) of Theorem 1 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 one, then \( D \) must also have a distance efficiency close to one.

The quality of design \( D \) is generally better than what is guaranteed by the lower bounds in Theorem 1. The equalities in parts (a) and (b) hold when \( B \) is equi-distant, but seldom hold otherwise. To see this, we observe that the \( L_p \)-distance of two rows of design \( D \) is a 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 \( d_H(A) \), but also render, by 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 hard to meet unless \( B \) is equi-distant or nearly equi-distant.
In Theorem 1, if \( d_{H,\text{eff}}(A) = d_{p,\text{eff}}(B) = 1 \), then \( d_{p,\text{eff}}(D) = 1 \), in which case, all three designs are equi-distant.

**Corollary 1.** If \( A \) and \( B \) in Theorem 1 are both equi-distant, then \( D \) is equi-distant and is thus a maximin distance design.

If we choose \( A \) to be a saturated orthogonal array \( OA(N, s^{n_1}) \), then it is equi-distant with \( d_H(A) = N/s \) according to Cheng (2014, Theorem 8.6). The Rao-Hamming construction gives a saturated \( OA(N, s^{n_1}) \) 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) constructed an \( LH(s, s) \), a Latin hypercube of \( s \) runs for \( s \) factors, can be constructed if \( 2s + 1 \) is a prime. We choose the next result is established.

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

Some designs from applying Theorem 2 are presented in Table 1.

The equi-distant maximin designs obtained 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 \( OA(s^k, s^{n_1}) \) with \( n_1 = (s^k - 1)/(s - 1) \) as \( A \).

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

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

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

---

**Table 1. Some equi-distant maximin designs \( U(N, s^n) \)**

<table>
<thead>
<tr>
<th>( A )</th>
<th>( B )</th>
<th>( D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( OA(9, 3^4) )</td>
<td>( LH(3, 3) )</td>
<td>( U(9, 3^{12}) )</td>
</tr>
<tr>
<td>( OA(27, 3^{11}) )</td>
<td>( LH(3, 3) )</td>
<td>( U(27, 3^{29}) )</td>
</tr>
<tr>
<td>( OA(81, 3^{40}) )</td>
<td>( LH(3, 3) )</td>
<td>( U(81, 3^{120}) )</td>
</tr>
<tr>
<td>( OA(243, 3^{21}) )</td>
<td>( LH(3, 3) )</td>
<td>( U(243, 3^{363}) )</td>
</tr>
<tr>
<td>( OA(729, 3^{64}) )</td>
<td>( LH(3, 3) )</td>
<td>( U(729, 3^{1092}) )</td>
</tr>
<tr>
<td>( OA(25, 5^6) )</td>
<td>( LH(5, 5) )</td>
<td>( U(25, 5^{30}) )</td>
</tr>
<tr>
<td>( OA(125, 5^{11}) )</td>
<td>( LH(5, 5) )</td>
<td>( U(125, 5^{55}) )</td>
</tr>
<tr>
<td>( OA(625, 5^{20}) )</td>
<td>( LH(5, 5) )</td>
<td>( U(625, 5^{100}) )</td>
</tr>
<tr>
<td>( OA(64, 8^5) )</td>
<td>( LH(8, 8) )</td>
<td>( U(64, 8^{25}) )</td>
</tr>
<tr>
<td>( OA(512, 8^{11}) )</td>
<td>( LH(8, 8) )</td>
<td>( U(512, 8^{64}) )</td>
</tr>
<tr>
<td>( OA(81, 9^{10}) )</td>
<td>( LH(9, 9) )</td>
<td>( U(81, 9^{90}) )</td>
</tr>
<tr>
<td>( OA(729, 9^{11}) )</td>
<td>( LH(9, 9) )</td>
<td>( U(729, 9^{99}) )</td>
</tr>
<tr>
<td>( OA(121, 11^{12}) )</td>
<td>( LH(11, 11) )</td>
<td>( U(121, 11^{122}) )</td>
</tr>
<tr>
<td>( OA(1331, 11^{133}) )</td>
<td>( LH(11, 11) )</td>
<td>( U(1331, 11^{1463}) )</td>
</tr>
</tbody>
</table>
Maximin distance designs

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

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>$d_{1,\text{eff}}(D)$</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>OA(16, 4⁵)</td>
<td>LH(4, 3)</td>
<td>U(16, 4¹⁵⁵)</td>
<td>0.800 Proposition 1</td>
</tr>
<tr>
<td>OA(64, 4¹ⁱ)</td>
<td>LH(4, 3)</td>
<td>U(64, 4⁶⁵)</td>
<td>0.800 Proposition 1</td>
</tr>
<tr>
<td>OA(256, 4¹⁷)</td>
<td>LH(4, 3)</td>
<td>U(256, 4²⁵⁵)</td>
<td>0.800 Proposition 1</td>
</tr>
<tr>
<td>OA(16, 5)</td>
<td>LH(5, 4)</td>
<td>U(25, 5²⁴)</td>
<td>0.875 Proposition 2</td>
</tr>
<tr>
<td>OA(125, 5³¹)</td>
<td>LH(5, 4)</td>
<td>U(125, 5¹²⁴)</td>
<td>0.875 Proposition 2</td>
</tr>
<tr>
<td>OA(625, 5¹⁵⁶)</td>
<td>LH(5, 4)</td>
<td>U(625, 5³²⁴)</td>
<td>0.875 Proposition 2</td>
</tr>
<tr>
<td>OA(49, 7²)</td>
<td>LH(7, 6)</td>
<td>U(49, 7¹⁶)</td>
<td>1 Proposition 2</td>
</tr>
<tr>
<td>OA(343, 7³⁷)</td>
<td>LH(7, 6)</td>
<td>U(343, 7³⁴²)</td>
<td>1 Proposition 2</td>
</tr>
<tr>
<td>OA(81, 9⁶⁰)</td>
<td>LH(9, 8)</td>
<td>U(81, 9⁶⁰)</td>
<td>0.900 Proposition 1</td>
</tr>
<tr>
<td>OA(729, 9¹⁰⁶)</td>
<td>LH(9, 8)</td>
<td>U(729, 9²⁸)</td>
<td>0.900 Proposition 1</td>
</tr>
<tr>
<td>OA(121, 11²⁸)</td>
<td>LH(11, 10)</td>
<td>U(121, 11¹²⁰)</td>
<td>0.975 Proposition 2</td>
</tr>
<tr>
<td>OA(169, 13¹⁴)</td>
<td>LH(13, 12)</td>
<td>U(169, 13¹⁶⁸)</td>
<td>0.929 Proposition 2</td>
</tr>
<tr>
<td>OA(256, 16¹⁷)</td>
<td>LH(16, 15)</td>
<td>U(256, 16²⁵⁵)</td>
<td>0.941 Proposition 1</td>
</tr>
<tr>
<td>OA(289, 17¹⁸)</td>
<td>LH(17, 16)</td>
<td>U(289, 17²⁸⁸)</td>
<td>0.979 Proposition 2</td>
</tr>
<tr>
<td>OA(361, 19²⁰)</td>
<td>LH(19, 18)</td>
<td>U(361, 19³⁶⁰)</td>
<td>0.958 Proposition 2</td>
</tr>
</tbody>
</table>

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

Propositions 1 and 2 together cover many $s$ values for a $U(N, s^n)$ 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 $d_{1,\text{eff}}(D) \geq 0.875$ while Proposition 2 gives a design with $d_{1,\text{eff}}(D) \geq 0.833$. When we examine the two LH(7, 6)s used in Propositions 1 and 2, we are pleasantly surprised to find that the LH(7, 6), constructed by Theorem 2 of Wang et al. (2018), is equi-distant. This means the design from Proposition 2 for $s = 7$ actually has $d_{1,\text{eff}}(D) = 1$. The equi-distant LH(7, 6) is displayed below:

```
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
```

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 LH($2^k, 2^k - 1$) with $L_2$-distance efficiency $d_{2,\text{eff}} \geq 1 - 1/2^k$ for any $k \geq 2$. We now use this design as $B$ while still taking a saturated orthogonal array as $A$.

Proposition 3. A $U(N, s^n)$, 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 $d_{2,\text{eff}}(D) \geq 1 - 1/2^{k_2}$ for all $k_1 \geq 2$ and $k_2 \geq 2$.

Taking $k_1 = 2$ and $k_2 = 3$ gives a $U(64, 8³⁶)$ with $d_{2,\text{eff}}(D) \geq 7/8$. If we take $k_1 = 2$ and $k_2 = 4$, we obtain a $U(256, 16¹³⁶)$ with $d_{2,\text{eff}}(D) \geq 15/16$. 


4. Further results

In the previous section after the main result in Theorem 1, our focus is the construction of design $D$ on condition that $A$ is a saturated orthogonal array, which is equi-distant under the Hamming distance. We shift gears in this section and consider the construction of design $D$ with $B$ chosen to be equi-distant 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 $\text{OA}(N, s^{n_1})$ 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 $\overline{A} = S \setminus A$. Let $\overline{A}$ and $A$ have $j$ and $n_1 - j$ columns, respectively. As $S$ is equi-distant with $d_H(S) = N/s - d_H^*(\overline{A})$, where $d_H^*(\overline{A})$ is the maximum Hamming distance between any two rows of $\overline{A}$. We are ready to present the next lemma.

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

$$d_{p, \text{eff}}(D) = d_{H, \text{eff}}(A) = \left(\frac{N - 1}{N}\right) \left(\frac{N - s d_H^*(\overline{A})}{N - js + j - 1}\right),$$

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

Obviously, for $j = 1, 2$, we have $d_H^*(\overline{A}) = j$ for any choice of $\overline{A}$ with $j$ columns. For $j \geq 3$, we have $d_H^*(\overline{A}) \leq j$. We therefore have that for $j \geq 1$,

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

which is close to 1, provided that $j$ is small.

For given $j$, we want to find an $\overline{A}$ that minimizes $d_H^*(\overline{A})$, thus maximizing $d_{p, \text{eff}}(D) = d_{H, \text{eff}}(A)$ owing to Lemma 1. We have seen already that any $\overline{A}$ minimizes $d_H^*(\overline{A})$ if $j = 1, 2$.

Although it may be too difficult to solve this optimization problem for all $j$, we are able to find a solution for certain $j$ values. This we present next.

The saturated orthogonal array $S$, the 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 $(u_1, \ldots, u_k)$ being equal to one. 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 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 $d_H^*(S_0) = N_0/s = s^{n_0 - 1}$. Taking $\overline{A} = S_0$ gives the main result of this section.

**Theorem 3.** Let $B$ be an equi-distant $U(s, q^{n_2})$ under the $L_p$-distance. Take $\overline{A} = S_0$ and thus $A = S \setminus S_0$. Then design $D$ in Theorem 1 is a $U(N, q^{(n_2 j - 1)n_2})$ that has

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

where $n_1 - n_0 = (s^k - s^{k_0})/(s - 1)$. This choice of $\overline{A}$ is optimal in that no other $\overline{A}$ can give a higher value of $d_{H, \text{eff}}(A)$.
columns are given in Yates order in Table 4, where column are supersaturated, deleting one column gives a U

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

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>D</th>
<th>(d_{1,\text{eff}}(D))</th>
<th>(d_{1,\text{eff}}(D_{-1}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>OA(9, 3⁴)</td>
<td>LH(3, 3)</td>
<td>U(9, 3⁴)</td>
<td>0.889</td>
<td>0.750</td>
</tr>
<tr>
<td>OA(27, 3⁶)</td>
<td>LH(3, 3)</td>
<td>U(27, 3²⁷)</td>
<td>0.963</td>
<td>0.917</td>
</tr>
<tr>
<td>OA(81, 3²⁷)</td>
<td>LH(3, 3)</td>
<td>U(81, 3³⁴)</td>
<td>0.988</td>
<td>0.972</td>
</tr>
<tr>
<td>OA(243, 3⁴¹)</td>
<td>LH(3, 3)</td>
<td>U(243, 3²⁴³)</td>
<td>0.996</td>
<td>0.991</td>
</tr>
<tr>
<td>OA(729, 3²⁴⁵)</td>
<td>LH(3, 3)</td>
<td>U(729, 3²⁷⁹)</td>
<td>0.999</td>
<td>0.997</td>
</tr>
<tr>
<td>OA(25, 5⁴)</td>
<td>LH(5, 5)</td>
<td>U(25, 5²⁵)</td>
<td>0.960</td>
<td>0.900</td>
</tr>
<tr>
<td>OA(125, 5²⁵)</td>
<td>LH(5, 5)</td>
<td>U(125, 5¹₂⁵)</td>
<td>0.992</td>
<td>0.980</td>
</tr>
<tr>
<td>OA(625, 5¹₂⁵)</td>
<td>LH(5, 5)</td>
<td>U(625, 5⁶²⁵)</td>
<td>0.998</td>
<td>0.996</td>
</tr>
<tr>
<td>OA(64, 8⁶)</td>
<td>LH(8, 8)</td>
<td>U(64, 8⁶⁴)</td>
<td>0.984</td>
<td>0.958</td>
</tr>
<tr>
<td>OA(512, 8⁶⁴)</td>
<td>LH(8, 8)</td>
<td>U(512, 8⁵¹²)</td>
<td>0.998</td>
<td>0.995</td>
</tr>
<tr>
<td>OA(81, 9⁶)</td>
<td>LH(9, 9)</td>
<td>U(81, 9⁶¹)</td>
<td>0.988</td>
<td>0.967</td>
</tr>
<tr>
<td>OA(729, 9²⁹)</td>
<td>LH(9, 9)</td>
<td>U(729, 9²⁹)</td>
<td>0.999</td>
<td>0.996</td>
</tr>
<tr>
<td>OA(121, 1¹₁)</td>
<td>LH(11, 11)</td>
<td>U(121, 1¹²¹)</td>
<td>0.992</td>
<td>0.977</td>
</tr>
<tr>
<td>OA(1331, 1¹¹²¹)</td>
<td>LH(11, 11)</td>
<td>U(1331, 1¹¹³³¹)</td>
<td>0.999</td>
<td>0.998</td>
</tr>
</tbody>
</table>

We now take B to be the LH(s, s) from Wang et al. (2018), which is equidistant under the L₁-distance.

**Corollary 2.** Let s be a prime power such that \(2s + 1\) is a prime. If we take B to be the LH(s, s) from Wang et al. (2018) and A to be S \ S₀ as in Theorem 3, then we obtain a U(N, sₙ) with \(n = s(s^k - s^{k₀})/(s - 1)\) with \(d_{1,\text{eff}}(D) = 1 - 1/N\). Of special interest is when \(k₀ = k - 1\), in which case, we obtain a U(N, sᴺ) with \(d_{1,\text{eff}}(D) = 1 - 1/N\).

**Example 1.** Take \(s = 3\) and \(k = 3\) in Corollary 2. Then design S is a saturated OA(27, 3¹³) whose set of 13 columns is

\[ S = \{ e₁, e₂, e₁e₂, e₁e²₃, e₁e₃, e₁e², e₂e₃, e₂e²₃, e₁e₂e₃, e₁e₂e²₃, e₁e₂e³, e₁e²e₃, e₁e²e³ \}. \]

If we take \(k₀ = 2\), then \(A = S₀ = \{ e₁, e₂, e₁e₂, e₁e₂e²₃ \}. \) So \(A = S \setminus A\) is an OA(27, 3⁶). Using this A in conjunction with an equi-distance LH(3, 3) as B, we obtain design D, a U(27, 3²⁷), with \(d_{1,\text{eff}}(D) = 26/27 = 0.963\). Design D is actually a maximin L₁-distance design because we can calculate that \(d₁(D) = 24\) and \(d_{1,\text{ave}}(D) = 324/13 < 25\), showing that the L₁-distance of D attains the upper bound \([324/13] = 24\).

Table 3 provides a list of U(N, sᴺ) that can be constructed by Corollary 2. These U(N, sᴺ)s are supersaturated, deleting one column gives a U(N, sᴺ⁻¹). In the last column of Table 3, we also provide the information on the distance efficiency for these U(N, sᴺ⁻¹)s.

Theorem 3 provides an optimal choice of \(A\) for a set of selected \(j\) values given by \(j = (s^k₀ - 1)/(s - 1)\) for \(k₀ = 1, \ldots, k - 1\), where \(j\) denotes the number of columns in \(A\). For any other \(j\) value, a sensible strategy of connecting the dots, as suggested by the same theorem, is to construct \(A\) by selecting the columns of S in Yates order. We use an example to illustrate the idea.

**Table 4. The 31 columns of OA(125, 5³¹)\)**

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>12</th>
<th>12²</th>
<th>12³</th>
<th>13</th>
<th>13²</th>
<th>13³</th>
<th>13⁴</th>
<th>23</th>
<th>23²</th>
<th>23³</th>
<th>23⁴</th>
<th>123</th>
</tr>
</thead>
<tbody>
<tr>
<td>12²</td>
<td>12³</td>
<td>12⁴</td>
<td>12⁵</td>
<td>13</td>
<td>13²</td>
<td>13³</td>
<td>13⁴</td>
<td>23</td>
<td>23²</td>
<td>23³</td>
<td>23⁴</td>
<td>123</td>
<td></td>
</tr>
</tbody>
</table>

**Example 2.** For \(s = 5\) and \(k = 3\), we have that S is a saturated OA(125, 5³¹) whose 31 columns are given in Yates order in Table 4, where column \(e₁^{u₁} \cdots eₖ^{uₖ}\) is replaced by \(1^{u₁} \cdots k^{uₖ}\) 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 $A$ in combination with the equi-distant \( LH(5, 5) \), we obtain a \( U(125, 5^m) \) for $n = 31, 30, 29, \ldots$. Table 5 gives the $L_1$-distance efficiencies of these designs for $n = 31, 30, \ldots, 19$.

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

<table>
<thead>
<tr>
<th>$5^n$</th>
<th>$d_{1, \text{eff}}(D)$</th>
<th>1</th>
<th>0.992</th>
<th>0.983</th>
<th>0.974</th>
<th>0.964</th>
<th>0.954</th>
<th>0.992</th>
<th>0.982</th>
<th>0.970</th>
<th>0.958</th>
<th>0.945</th>
<th>0.930</th>
<th>0.914</th>
</tr>
</thead>
<tbody>
<tr>
<td>155</td>
<td>1</td>
<td>0.992</td>
<td>0.983</td>
<td>0.974</td>
<td>0.964</td>
<td>0.954</td>
<td>0.992</td>
<td>0.982</td>
<td>0.970</td>
<td>0.958</td>
<td>0.945</td>
<td>0.930</td>
<td>0.914</td>
<td></td>
</tr>
</tbody>
</table>

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

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

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

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

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.

If $B$ is equi-distant, then $(B, B)$ is also equi-distant. Therefore, designs with larger $n$ for given $N$ are readily available. We next provide in Table 7 a comparison of our designs with those constructed by R packages SLHD (Ba 2013) and lhs (Carnell, 2020) in the range $N/5 \leq n < N$. 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, 5. We see that the R package SLHD is always better than the R package lhs. Our method underperforms the SLHD for small $n$ values and becomes dominant as soon as the ratio $N/n$ exceeds certain threshold. This phenomenon also occurs for designs of other $N$ values that are not reported here due to space consideration.

He (2019) introduced interleaved lattice-based maximin $L_2$-distance designs and provided an R package \texttt{InterleavedMaximinD} for the implementation of his method. Table 8 compares our designs obtained from Proposition 3 with the designs generated by \texttt{InterleavedMaximinD} for $N = 256$ and $n = 136$ and 170 under both $L_2$- and $L_1$-distances. For reference, corresponding designs from R package SLHD are also included. We see that the designs from InterleavedMax-
Maximin distance designs

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

<table>
<thead>
<tr>
<th>$N$</th>
<th>$s$</th>
<th>$n$</th>
<th>25</th>
<th>25</th>
<th>30</th>
<th>35</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>80</th>
<th>124</th>
</tr>
</thead>
<tbody>
<tr>
<td>125</td>
<td>5</td>
<td>Ours</td>
<td>2.5</td>
<td>2.5</td>
<td>5.0</td>
<td>7.5</td>
<td>10.0</td>
<td>15.0</td>
<td>20.0</td>
<td>30.0</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SLHD</td>
<td>6.2</td>
<td>6.2</td>
<td>7.7</td>
<td>9.3</td>
<td>11.2</td>
<td>14.5</td>
<td>17.7</td>
<td>24.1</td>
<td>39.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LHS</td>
<td>4.5</td>
<td>4.5</td>
<td>5.7</td>
<td>7.3</td>
<td>8.2</td>
<td>11.1</td>
<td>13.7</td>
<td>19.1</td>
<td>32.0</td>
</tr>
<tr>
<td>343</td>
<td>7</td>
<td>Ours</td>
<td>13.3</td>
<td>16.0</td>
<td>21.3</td>
<td>26.7</td>
<td>34.7</td>
<td>42.7</td>
<td>58.7</td>
<td>82.7</td>
<td>131.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SLHD</td>
<td>16.1</td>
<td>17.7</td>
<td>21.1</td>
<td>24.6</td>
<td>31.1</td>
<td>36.4</td>
<td>47.4</td>
<td>65.5</td>
<td>101.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LHS</td>
<td>12.8</td>
<td>14.2</td>
<td>17.2</td>
<td>20.6</td>
<td>25.3</td>
<td>30.1</td>
<td>40.4</td>
<td>57.5</td>
<td>91.2</td>
</tr>
</tbody>
</table>

iminD are the best under the $L_2$-distance and our designs are the best under the $L_1$-distance. The superior performance of InterleavedMaximinD under the $L_2$-distance appears to be because the corresponding designs only have 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 often does not generate designs with required run sizes. For examples, InterleavedMaximinD outputs a design with 128 runs when inputting $N = 64$ and $n = 36$ and a design with 567 runs when inputting $N = 512$ and $n = 192$.

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

Table 8. Comparisons with designs from InterleavedMaximinD in terms of both $L_1$-distance $d_1$ and $L_2$-distance $d_2$

<table>
<thead>
<tr>
<th>$N$</th>
<th>$n$</th>
<th>design</th>
<th>$s$</th>
<th>$d_2$</th>
<th>$d_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>136</td>
<td>SLHD</td>
<td>256</td>
<td>17.5</td>
<td>37.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ours</td>
<td>16</td>
<td>24.2</td>
<td>38.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lattice</td>
<td>2</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>256</td>
<td>170</td>
<td>SLHD</td>
<td>256</td>
<td>22.6</td>
<td>48.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ours</td>
<td>4</td>
<td>35.6</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lattice</td>
<td>2</td>
<td>37</td>
<td>37</td>
</tr>
</tbody>
</table>

SLHD: R package SLHD; Ours: our designs; Lattice: R package InterleavedMaximinD.

5. Concluding remarks

Constructing designs with large distances is the central theme of the paper. Besides having large distances, our constructed designs also enjoy some other space-filling properties. It is obvious that, given $A$ is an orthogonal array, if $B$ has orthogonal columns, then design $D$ must also have orthogonal columns. Our designs also 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 design $D$ is space-filling in most of two dimensions.

There are a number of further issues that could lead to interesting work. The current paper mainly looks at designs constructed when at least one of $A$ and $B$ is equi-distant. 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 equi-distant, level permutation of \( A \) has an effect on the distance of \( D \), a point alluded to, though somewhat implicitly, in the discussion after Theorem 1, and this effect becomes more pronounced if \( B \) is far from being equi-distant. This important scenario would be very interesting to explore in the future.

**ACKNOWLEDGEMENT**

Research of Boxin Tang is supported by the Natural Sciences and Engineering Research Council of Canada. Research of Wenlong Li and Min-Qian Liu is supported by the National Natural Science Foundation of China, and National Ten Thousand Talents Program of China, Tianjin Development Program for Innovation and Entrepreneurship and Tianjin “131” Talents program. The authorship is 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 \( a_i \) and \( a_j \) be the corresponding two rows of \( A \). A moment of thought gives that

\[
d_p(c_i, c_j) \geq d_H(a_i, a_j)d_p(B) \geq d_H(A)d_p(B).
\]

This shows that \( d_p(D) \geq d_H(A)d_p(B) \). Part (a) of Theorem 1 is established.

To prove part (b), we recall from Section 2 the average distance for a \( U(N, q^n) \) given in (1).

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

we obtain \( d_{p,ave} = nNW(q)/(N - 1) \) where \( W(q) = 2 \sum_{i=1}^{q-1} i^p(q-i)/q^2 \). Applying this to design \( B \) and \( D \), we obtain \( d_{p,ave}(B) = n_2 sW(q)/(s - 1) \) and \( d_{p,ave}(D) = (n_1 n_2) N W(q)/(N - 1) \). For design \( A \), we have from (3) in Section 2 that \( d_{H,ave}(A) = N(s - 1)/((N - 1)s) \). We therefore have that

\[
d_{p,ave}(D) = d_{H,ave}(A) d_{p,ave}(B).
\]

This implies that \( d_{p,eff}(D) = d_{p}(D)/d_{p,ave}(D) \geq d_{H,eff}(A) d_{p,eff}(B) \) by part (a) of Theorem 1 and the definitions of \( d_{H,eff}(A) \) in (4) and \( d_{p,eff}(B) \) in (2). The last statement in Theorem 1 is obvious from the proof of part (a).

**Proof of Lemma 1**

Since \( B \) is equi-distant under the \( L_p \)-distance, we have that \( d_{p,eff}(B) = 1 \). Therefore \( d_{p,eff}(D) = d_{H,eff}(A) \). We also know that \( d_{H}(A) = N/s - d_{L_p}(\overline{A}) \).

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

\[
d_{p,eff}(D) = d_{H,eff}(A) = d_{H}(A)/d_{H,ave}(A) = \frac{N - 1}{N} \left( \frac{N - s d_{L_p}(\overline{A})}{N - js + j - 1} \right),
\]

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

**Proof of Theorem 3**

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

**REFERENCES**

Maximin distance designs