Journal of Statistical Planning and Inference 222 (2023) 209–225

Contents lists available at ScienceDirect

Journal of Statistical Planning and Inference

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

Construction of uniform projection designs via level permutation and expansion

Yishan Zhou [a], Qian Xiao [b], Fasheng Sun [a,*]

[a] Department of Statistics, KLAS and School of Mathematics and Statistics, Northeast Normal University, Changchun, 130024, China
[b] Department of Statistics, University of Georgia, 310 Herty Dr, Athens, GA, 30602, USA

ARTICLE INFO

Article history:
Received 17 July 2021
Received in revised form 22 June 2022
Accepted 23 June 2022
Available online 28 June 2022

MSC:
primary 62K15
secondary 62K99

Keywords:
Computer experiment
Fractional factorial design
Generalized minimum aberration
Latin hypercube design
Orthogonal array

ABSTRACT

Computer experiments are widely used in modern science and industrial applications that require space-filling designs. Uniform projection designs (UPDs) have recently been proposed to address designs' space-filling properties for low-dimensional projections. UPDs are desirable for experiments in which only portions of factors are active. The construction of UPDs with flexible sizes is challenging, especially for large ones. In this paper, we systematically study the construction methods of UPDs via level permutation and/or level expansion. For each approach, we establish theoretical results connecting the uniform projection properties of the generated designs with the properties of the corresponding initial designs. Based on the established theoretical results, efficient algorithms are developed to construct UPDs with flexible sizes, which leads to many practically useful designs.

1. Introduction

Computer experiments are widely used to emulate complex physical systems (Santner et al., 2003; Fang et al., 2006; Garud et al., 2017; Gramacy, 2020; Lukemire et al., 2021). Space-filling designs whose points are allocated evenly in the experimental regions are recommended for computer experiments (Fang et al., 2006; Gramacy, 2020). Space-filling Latin hypercube designs (LHDs) and fractional factorial designs (FFDs) are popular (Joseph, 2016; Lin and Tang, 2015; Xiao and Xu, 2018; Xiao et al., 2019). An LHD is an $n \times k$ matrix whose columns are permutations of numbers 1 to n (McKay et al., 1979). LHDs have unique point projections, thus having no replication, on each dimension.

In the current literature, the maximin distance criterion (Johnson et al., 1990) and the discrepancy criteria (Hickernell, 1998) are two popular space-filling measures. The former seeks to maximize the minimum inter-site distances among the design points (Lin and Tang, 2015; Sun and Tang, 2017a,b; Wang et al., 2018; Xiao and Xu, 2017, 2018; Li et al., 2020). The latter aims to minimize some discrepancy criteria, including the centered L_2-discrepancy (CD), the wrap-around L_2-discrepancy (WD) and the mixture discrepancy (MD) (Zhou et al., 2013). Optimal designs under either criterion focus on the space-filling properties over the entire design spaces, but they may have poor projection uniformity in low dimensions (Joseph et al., 2015).

In many computer experiments, only a few out of the numerous factors are active (Kleijnen, 2017; Moon et al., 2012; Woods and Lewis, 2016). Thus, an appropriate design should be space-filling not only in the full-dimensional space but

* Corresponding author.
E-mail address: sunfs359@nenu.edu.cn (F. Sun).

https://doi.org/10.1016/j.jspi.2022.06.010

also over all low-dimensional projections. Under this consideration, Joseph et al. (2015) proposed the maximum projection (Maxpro) design, and Sun et al. (2019) proposed the uniform projection design (UPD), where the former considers a distance metric and the latter relies on a discrepancy measure. Maxpro designs assume that all sub-spaces are equally important, while UPDs focus more on lower-dimensional projections. UPDs have the smallest average CD values of all two-dimensional projections and are shown to have good space-filling properties over all sub-spaces in terms of the distance, uniformity and orthogonality (Sun et al., 2019; Wang et al., 2020).

To find an n-run, k-factor and s-level UPD, the entire search space includes as many as $(n!/((n/s)!)^s)^k$ candidate designs (including isomorphism). Clearly, for UPDs of large sizes, a direct search over the entire space can be inefficient. In the current literature, the procedures of level permutation and level expansion are widely used to restrict the search space for identifying optimal designs (Tang, 1993; Leary et al., 2003; Tang et al., 2012; Zhou and Xu, 2014; Jiang and Ai, 2017; Xiao and Xu, 2018). A key problem in such methods is to identify appropriate initial designs which will determine the properties of sub-spaces for searching.

In this paper, we propose to construct UPDs via level permutation (LP), level expansion (LE), both level permutation and expansion (BLPE), and step-by-step level permutation and expansion (SLPE) according to the required design sizes. Theoretical results are established to identify the "average-best" sub-spaces for searching. Specifically, for all these four construction methods, we connect the average uniform projection properties of the generated designs with the uniform projection properties, distance structures and generalized word-length patterns of their corresponding initial designs. Initial designs with small A_2 values and small ϕ values should be used justified by both theoretical and empirical results. A tailored threshold accepting global optimization algorithm is developed for searching UPDs. Guidelines on when to apply these four constructions are discussed in detail. For moderate or large UPDs, SLPE is generally recommended as illustrated in Section 4.

The rest of this paper is organized as follows. Section 2 introduces the notation and preliminaries. Section 3 shows the theoretical results for the proposed constructions. Section 4 discusses the construction guidelines and show some numerical studies. Section 5 concludes and discusses some future work. All proofs and technical details are given in Appendix A.

2. Notation and preliminaries

Denote an n-run, k-factor and s-level (labeled as $1, 2, \ldots, s$) design as (n, s^k). A design is an orthogonal array (OA) of strength t, denoted as $OA(n, k, s, t)$, if all possible level combinations appear the same number of times in its every $n \times t$ sub-matrix (Hedayat et al., 1999). In practice, researchers often focus on OAs of strength $t = 2$. A design is balanced if every level appears the same number of times in its every column. Specifically, a Latin hypercube design, denoted as $LHD(n, k)$, is a balanced (n, n^k) design. Throughout this paper, we focus on balanced designs.

To evaluate designs' aliasing structures, Xu and Wu (2001) introduced the generalized word length pattern (GWLP). For a design $D(n, s^k)$, consider the full ANOVA model $Y = X_0\alpha_0 + X_1\alpha_1 + \cdots + X_n\alpha_n + \epsilon$, where Y is the response vector, α_0 is the intercept, X_0 is an $n \times 1$ vector of all 1's, α_j is an $(s-1)^j\binom{k}{j} \times 1$ vector including all jth-order factorial effects, X_j is an $n \times (s-1)^j\binom{k}{j}$ matrix consisting of jth factor contrast coefficients ($j = 1, \ldots, k$), and $\epsilon \sim N(0, \sigma^2)$ is a random error. Xu and Wu (2001) defined $A_j(D) = n^{-2}|X_0^T X_j|^2$ to measure the overall aliasing between the intercept and all jth-order factorial effects, where $|X|^2 = \mathrm{tr}(X^T X)$ and $j = 0, \ldots, k$. It is straightforward to show that all balanced designs D satisfy $A_0(D) = 1$. The GWLP of design D is the vector $(A_1(D), \ldots, A_k(D))$. Xu and Wu (2001) proposed to sequentially minimize designs' GWLPs. A design D is an OA of strength t if and only if $A_1(D) = \cdots = A_t(D) = 0$.

For an (n, s^k) design $D = (x_{il})_{n\times k}$, let $x_i = (x_{i1}, \ldots, x_{ik})$ and $x_j = \left(x_{j1}, \ldots, x_{jk}\right)$ be its ith and jth rows, respectively. Denote the Hamming distance between rows x_i and x_j as $h_{i,j}$, which is the number of positions at which the two rows are different. Let $d_p\left(x_i, x_j\right) = \left(\sum_{l=1}^k \left|x_{il} - x_{jl}\right|^p\right)^{1/p}$ be the L_p-distance between two rows x_i and x_j. Let $d_p(D) = min\{d_p(x_i, x_j), 1 \le i < j \le n\}$ be the L_p-distance of design D. The maximin L_p-distance design maximizes the value of $d_p(D)$ among all designs of the same size. In this paper, we focus on the L_1-distance measure (a.k.a., Manhattan distance); that is, we use $d_{i,j} = \sum_{l=1}^k d_{il,jl}$ where $d_{il,jl} = \left|x_{il} - x_{jl}\right|$.

The centered L_2-discrepancy (CD) proposed by Hickernell (1998) is a widely used criterion for measuring designs' space-filling properties. It has a clear geometric interpretation that the number of points in any chosen rectangular space should be proportional to the volume of the chosen space if the design points are space-filling in the whole space or sub-spaces. Based on the CD metric, Sun et al. (2019) proposed the uniform projection criterion which is defined as

$$\phi(D) = \frac{2}{k(k-1)} \sum_{|u|=2} CD(D_u),$$

where u is a subset of $\{1, 2, \ldots, k\}$, $|u|$ is the cardinality of u, and D_u is the projection of D onto the dimensions indexed by the elements in u. A uniform projection design (UPD) minimizes the value of $\phi(D)$ among all possible designs of the same size. Sun et al. (2019) showed that

$$\phi(D) = \frac{g(D)}{4k(k-1)n^2 s^2} + C(k, s), \tag{1}$$

where $g(D) = \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - 2\sum_{i=1}^{n}(\sum_{j=1}^{n} d_{i,j})^2/n$, and the constant $C(k,s) = (4(5k-2)s^4 + 30(3k-5)s^2 + 15k + 33)/(720(k-1)s^4) + (1+(-1)^s)/(64s^4)$. Eq. (1) provides a fast way to compute the $\phi(D)$ values via calculating the pairwise L_1-distances in D.

Sun et al. (2019) gave a lower bound of $\phi(D)$, and Wang et al. (2020) gave an improved lower bound and a new upper bound. We summarize their findings in the following lemma.

Lemma 1. *For any balanced (n, s^k) design D, we have $max\{\phi_{LB1}, \phi_{LB2}\} \leqslant \phi(D) \leqslant \phi_{UB}$, where*

$$\phi_{LB1} = \frac{5k(4s^4 + 2(13n-17)s^2 - n + 5) - (n-1)(8s^4 + 150s^2 - 33)}{720(n-1)(k-1)s^4} + \frac{1+(-1)^s}{64s^4},$$

$$\phi_{LB2} = \frac{26s^2 - 1}{144s^4} + \frac{1+(-1)^s}{64s^4},$$

$$\phi_{UB} = \frac{(10k-8)s^4 + (140k-150)s^2 - 25k + 33}{720(k-1)s^4} + \frac{1+(-1)^s}{64s^4}.$$

The lower bound ϕ_{LB1} is achieved if and only if D is an L_1-equidistant design. The lower bound ϕ_{LB2} is achieved if and only if D is an OA.

Wang et al. (2020) defined the relative ϕ-efficiency of a design D as

$$\phi_{RE}(D) = \frac{\phi_{UB} - \phi(D)}{\phi_{UB} - \phi_{LB}}, \tag{2}$$

where $\phi_{LB} = max\{\phi_{LB1}, \phi_{LB2}\}$ and ϕ_{UB} are given in Lemma 1. Clearly, we have $0 \leqslant \phi_{RE}(D) \leqslant 1$, and larger $\phi_{RE}(D)$ values indicate better projection uniformity of designs.

Next, we define the procedures of level permutation (LP) and level expansion (LE). Starting from an (n, s^k) initial design D, we can randomly permute the s levels in its one or more factors to generate a new design D' of the same size, which is called the LP procedure. From any initial design D, we have $(s!)^k$ possible D''s generated via LP. In addition, starting from a low-level initial design $D(n, s^k)$, we can generate high-level designs $D'(n, (ms)^k)$ via LE; that is, for each column in D, we replace the n/s entries of level l $(l = 1, 2, \ldots, s)$ with $n/(ms)$ replicates of random permutations of $\{(l-1)m+1, (l-1)m+2, \ldots, (l-1)m+m\}$, where n, k, s and m are all integers larger than 1 and n is divisible by ms. Specifically, the D''s are LHDs if $m = n/s$. From any initial design D, we have $((n/s)!/(r!)^m)^{sk}$ possible D''s generated via LE, where $r = n/(ms)$.

In the procedure of both level permutation and expansion (BLPE), we first perform LP to an initial design $D(n, s^k)$, and then for each generated design via LP we perform LE to obtain the generated designs $D'\left(n, (ms)^k\right)$. Clearly, from any initial design D, there are in total $(s!)^k\,((n/s)!/(r!)^m)^{sk}$ possible D''s via BLPE. Note that this number is much smaller than the total number of possible designs with n runs, k factors and ms levels which is $(n!/((n/ms)!)^{ms})^k$. Obviously, the LP, LE and BLPE procedures restrict the whole search space to some much smaller sub-spaces. The initial designs D determine which sub-spaces to search over, and we want to choose the best D that will lead to the "average-best" performances of all the generated designs.

Example 1. To illustrate the LP procedure, we consider replacing the levels $(1, 2, 3, 4)$ in both columns of an initial design $D_0(8, 4^2)$ with a random permutation $(1, 3, 2, 4)$ to generate a new design D, where

$$D_0 = \begin{pmatrix} 1 & 1 & 3 & 3 & 2 & 2 & 4 & 4 \\ 1 & 2 & 3 & 4 & 1 & 2 & 3 & 4 \end{pmatrix}^T, \quad D = \begin{pmatrix} 1 & 1 & 2 & 2 & 3 & 3 & 4 & 4 \\ 1 & 3 & 2 & 4 & 1 & 3 & 2 & 4 \end{pmatrix}^T.$$

Design D_0 has a uniform projection value of $\phi(D_0) = 0.02$ and a relative ϕ-efficiency of $\phi_{RE}(D_0) = 47\%$, while design D has $\phi(D) = 0.01$ and $\phi_{RE}(D) = 94\%$. Clearly, we can improve the design's projection uniformity by choosing the best out of the 576 possible generated designs D via LP.

To illustrate the LE procedure, we consider generating an LHD D' from the above D after LP. For each column in D, we replace all entries of 1 (2, 3 or 4) with a random permutation of numbers: $(1, 2)$ ($(3, 4)$, $(5, 6)$ or $(7, 8)$). Here are two possible generated LHDs:

$$D_2' = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 5 & 3 & 7 & 2 & 6 & 4 & 8 \end{pmatrix}^T, \quad D_3' = \begin{pmatrix} 2 & 1 & 3 & 4 & 5 & 6 & 8 & 7 \\ 2 & 6 & 4 & 8 & 1 & 5 & 3 & 7 \end{pmatrix}^T.$$

We have $\phi(D_2') = 0.007$ ($\phi_{RE}(D_2') = 77\%$) and $\phi(D_3') = 0.004$ ($\phi_{RE}(D_3') = 90\%$). Clearly, we can improve the design's projection uniformity by choosing the best out of the 256 possible generated designs D' via LE.

In addition, to illustrate the impacts of the initial designs on the generated designs in LP or LE, we compare pairs of initial designs having the same design sizes but different ϕ values. In Table 1, we list the minimum and average ϕ values (defined in (1)) and the corresponding ϕ-efficiency ϕ_{RE} (defined in (2)) of the generated designs searched in the same time, where we separate the results from the pair of initial designs using a slash. We can observe that initial designs with smaller ϕ values tend to generated designs with smaller ϕ values.

Table 1
The ϕ-values (multiplied by 10^4) and ϕ-efficiencies (%) of designs generated from pairs of initial designs via either LP or LE.

	(n, s^k)	$\phi(D)$	$\phi(D')(\phi_{RE})$	
			$\phi_{min}(\phi_{RE\_min})$	$\bar{\phi}(\phi_{RE\_ave})$
LP	$(27, 3^5)$	199.76/308.13	199.76(100)/304.01(32.45)	199.76(100)/304.13(32.37)
	$(50, 5^5)$	72.11/131.63	72.11(100)/121.71(66.84)	72.11(100)/124.12(65.23)
LE	$(27, 27^5)$	199.76/308.13	5.02(98.26)/56.43(63.04)	5.78(97.73)/56.97(62.67)
	$(50, 50^5)$	72.11/131.63	1.59(99.41)/39.77(73.23)	1.87(99.21)/39.92(73.13)

Table 2
The ϕ-values (multiplied by 10^4) of designs generated from five 2^{10-3} initial designs via LP.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Ave	True.Ave
(0,0,0,3)	466.58	**466.58**	**466.58**	**466.58**
(0,0,1,2)	466.58	**466.58**	**466.58**	**466.58**
(0,1,0,2)	470.05	470.05	470.05	470.05
(0,2,0,1)	473.52	473.52	473.52	473.52
(0,3,0,3)	477.00	477.00	477.00	477.00

In next section, we will give theoretical results on how to choose initial designs to improve the overall searching efficiency.

3. Theoretical results

In this section, we systematically study the theoretical properties for constructing UPDs under three scenarios: level permutation (LP), level expansion (LE) and both level permutation and expansion (BLPE). Numerical examples are given to illustrate the theoretical results.

3.1. Designs generated via LP

For an (n, s^k) initial design D, let $\mathcal{P}(D)$ be the collection of all designs D' generated by permuting the levels of D. Let $\overline{\phi}_{\mathcal{P}}\left(D'\right)$ be the average ϕ values of all designs in $\mathcal{P}(D)$.

Theorem 1. *For any balanced (n, s^k) design D, when all possible level permutations of D are considered, we have*

$$\overline{\phi}_{\mathcal{P}}\left(D'\right) = \frac{1}{n_{\mathcal{P}}} \sum_{D' \in \mathcal{P}(D)} \phi(D') = \frac{(s+1)^2}{18k(k-1)s^4} A_2(D) + \phi_{LB2},$$

where the constant $n_{\mathcal{P}} = (s!)^k$ and ϕ_{LB2} is defined in Lemma 1.

Theorem 1 shows that the average ϕ values of all designs in $\mathcal{P}(D)$ are a linear function of the initial design D's A_2 (in GWLP) value. Thus, initial designs with small A_2 values (e.g., OAs) are preferred in LP, which will lead to the average-best (smallest average ϕ values) sub-spaces for searching. As LP will not change the ϕ value of any two-level design, we can prove the following corollary.

Corollary 1. *For a balanced $(n, 2^k)$ design D, $\phi(D) = \frac{1}{32k(k-1)} A_2(D) + \frac{215}{4608}$.*

By Corollary 1, it is seen that minimizing $\phi(D)$ is equivalent to minimizing its A_2 value for two-level designs.

Example 2. Consider five 2^{10-3} and five 3^{7-2} fractional factorial designs with different GWLPs, from which we enumerate all possible 1024 and 279 936 generated designs D' via LP respectively. We show their minimum ("Min" column) and average ("Ave" column) ϕ values in Tables 2 and 3, respectively. Additionally, we show the true average ("True.Ave" column) ϕ values of all possible generated designs in $\mathcal{P}(D)$ by Theorem 1. We use bold font to show the best results throughout the paper.

From Table 2, it is seen that an $OA(n, 2^k)$ $(A_2 = 0)$ is a UPD with $\phi(D) = 215/4068$, which illustrates Corollary 1. From Tables 2 and 3, it is not difficult to find that the average ϕ values calculated based on Theorem 1 is equal to the average ϕ values for all generated designs enumerated, which verifies Theorem 1. We can see that the ϕ values of generated designs are different when using different initial designs which have different GWLPs. Specifically, an initial design having a smaller A_2 value will lead to a better search space of generated designs in terms of the average uniform projection properties, which illustrates Theorem 1. Note that initial designs D with $A_2 = 0$ will have the same $\phi(D)$ value which achieves a lower bound as described by Lemma 1.

Table 3
The ϕ-values (multiplied by 10^4) of designs generated from five 3^{7-2} initial designs via LP.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Ave	True.Ave
(0,0,2,6)	199.76	**199.76**	**199.76**	**199.76**
(0,0,4,2)	199.76	**199.76**	**199.76**	**199.76**
(0,0,8,0)	199.76	**199.76**	**199.76**	**199.76**
(0,2,0,4)	206.29	204.33	204.99	204.99
(0,4,0,4)	212.82	208.90	210.21	210.21

3.2. Designs generated via LE

Starting from a low-level design $D(n, s^k)$, let $\mathcal{E}(D)$ be the collection of all high-level generated designs $D'(n, (ms)^k)$ via LE, where n, k, s and m are all integers larger than 1 and n is divisible by ms. Let $\overline{\phi}_{\mathcal{E}}\left(D'\right)$ be the average ϕ values of all designs in $\mathcal{E}(D)$.

Theorem 2. *For a balanced (n, s^k) design D, when all possible level expansions of D are considered, we have*

$$\overline{\phi}_{\mathcal{E}}\left(D'\right) = \frac{1}{n_{\mathcal{E}}} \sum_{D' \in \mathcal{E}(D)} \phi(D') = \phi(D) - \frac{m^2-1}{6k(k-1)m^2s^2n(n-s)} \sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j}d_{i,j} + \frac{n^2\left(m^2-1\right)^2}{18k(k-1)m^4s^4(n-s)^2} A_2(D) + C_1, \quad (3)$$

where the constant $n_{\mathcal{E}} = ((n/s)!/(r!)^m)^{sk}$, $r = n/(ms)$, $m = n/s$ and

$$\begin{aligned} C_1 = &\frac{\left(m^2-1\right)\left[-(4kn-3)s^3 + 2(2kn-3)ns^2 + (2k+3n+2)ns - 4kn^2\right]}{72(k-1)m^2s^3(n-s)^2} \\ &+ \frac{n\left(m^2-1\right)\left(s^2 + n\left(m^2-1\right)\right)}{36m^4s^4(n-s)^2} - \frac{1+(-1)^s}{64s^4} - \frac{\left(m^2-1\right)\left(2m^2s^2 - m^2 - 1\right)}{288m^4s^4} \\ &- \frac{\left(m^2-1\right)\left(2(11k-9)m^2s^2 - 3(k-1)m^2 - 3k + 3\right)}{96(k-1)m^4s^4} + \frac{1+(-1)^{ms}}{64m^4s^4}. \end{aligned}$$

Theorem 2 connects the generated designs' average uniform projection properties with the initial design's uniform projection property, distance structure and A_2 value (in GWLP). Next, we identify the dominant term of $\overline{\phi}_{\mathcal{E}}\left(D'\right)$.

Lemma 2. *For a balanced design $D(n, s^k)$, the ϕ criterion defined in (1) is equivalent to*

$$\phi(D) = \frac{1}{4k(k-1)n^2s^2} \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - \frac{1}{k(k-1)ns^4} \sum_{i=1}^{n} \sum_{1 \leqslant p < q \leqslant k} \left(x_{ip} - s_0\right)^2 \left(x_{iq} - s_0\right)^2 + C_2, \quad (4)$$

where $C_2 = C(k, s) - \left(s^2-1\right)\left((25k+3)s^2 - 25k - 7\right)/480(k-1)s^4$ is a constant with $C(k, s)$ defined in (1) and $s_0 = (s+1)/2$.

Remark 1. The first term of $\phi(D)$ in (4) is always greater than the absolute value of its second term; that is,

$$\frac{1}{4k(k-1)n^2s^2} \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 > \frac{1}{k(k-1)ns^4} \sum_{i=1}^{n} \sum_{1 \leqslant p < q \leqslant k} \left(x_{ip} - s_0\right)^2 \left(x_{iq} - s_0\right)^2.$$

By Lemma 2 and Remark 1, it is seen that the ϕ value of a design D is dominated by $\sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 / \left[4k(k-1)n^2s^2\right]$. Thus, we have the following Remark 2 which reveals the dominated term in Theorem 2.

Remark 2. Denote the dominated term of $\phi(D)$ as $T_1 = \frac{1}{4k(k-1)n^2s^2} \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2$. Denote the absolute values of the second and the third terms in (3) as $T_2 = \frac{m^2-1}{6k(k-1)m^2s^2n(n-s)} \sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j}d_{i,j}$ and $T_3 = \frac{n^2(m^2-1)^2}{18k(k-1)m^4s^4(n-s)^2} A_2(D)$, respectively. In Theorem 2, we have:

(1) T_1 is not less than T_2, where $T_1/T_2 \geqslant 1 + \frac{(s-2)(s-1)}{2k(s+1)}$;
(2) T_2 is not less than T_3, when $A_2(D) \leqslant 4k$.

Table 4
Comparisons between the absolute values of the first two terms in Lemma 2 and the second term in Theorem 2.

s	Initial design	m	T_1	T_2	T_4	T_1/T_2	$T_1/(T_2+T_4)$
2	$(4, 2^3)$	2	0.03125	0.03125	0.00195	1.00	0.94
	$(8, 2^5)$	4	0.02344	0.01953		1.20	1.09
	$(128, 2^{10})_1$	64	0.01944	0.01317		1.48	1.29
	$(128, 2^{10})_2$		0.01979	0.01340		1.48	1.29
3	$(9, 3^4)$	3	0.03429	0.02195	0.00274	1.56	1.39
	$(27, 3^{13})$	3	0.02503	0.01235		2.03	1.66
		9		0.01372		1.83	1.52
	$(243, 3^7)_1$	81	0.02884	0.01415	0.00281	2.04	1.70
	$(243, 3^7)_2$		0.02956	0.01442	0.00287	2.05	1.71
5	$(25, 5^6)$	5	0.03360	0.01280	0.00320	2.63	2.10
7	$(49, 7^8)$	7	0.05702	0.01737	0.00333	3.28	2.66
9	$(81, 9^{10})$	3	0.03167	0.00610	0.00339	5.19	3.34
		9		0.00677		4.68	3.12

Note: $(128, 2^{10})_1$ refers to the 2^{10-3} design with $(A_1, A_2, A_3, A_4) = (0, 1, 0, 2)$; $(128, 2^{10})_2$ refers to the 2^{10-3} design with $(0, 2, 0, 1)$; $(243, 3^7)_1$ refers to the 3^{7-2} design with $(0, 2, 0, 4)$; $(243, 3^7)_2$ refers to the 3^{7-2} design with $(0, 4, 0, 4)$; and other designs are OAs with $t = 2$.

Table 5
Comparisons between the absolute values of the second and third terms in Theorem 2.

(A_1, A_2, A_3, A_4)	s	m	$4k$	T_2	T_3	T_2/T_3
(0,1,0,2)	2	64	40	0.01317	0.00004	330.83
(0,2,0,1)				0.01340	0.00008	168.37
(0,3,0,3)				0.01364	0.00012	114.22
(0,2,0,4)	3	81	28	0.01415	0.00002	422.78
(0,4,0,4)				0.01442	0.00007	215.34

By Remark 2, the average uniform projection property of all generated designs $\overline{\phi}_{\mathcal{E}}\left(D'\right)$ in Theorem 2 is dominated by the initial design's uniform projection property $\phi(D)$. Note that the difference between the terms T_1 and T_2 increases quickly as s increases; see the numerical results in Table 4 for illustration. Intuitively, this is mainly because the difference between $h_{i,j}$ and $d_{i,j}$ becomes larger for higher levels. For clarity, denote the absolute value of the second term in (4) as

$$T_4 = \sum_{i=1}^{n} \sum_{1 \leqslant p<q \leqslant k} \left(x_{ip} - s_0\right)^2 \left(x_{iq} - s_0\right)^2 / k(k-1)ns^4.$$

In Table 4, it is seen that the difference between the terms T_1 and T_4 also increases quickly as s increases. In addition, since $A_2(D) \leqslant 4k$ clearly holds in most practical cases, T_2 is not less than T_3 by Remark 2; see Table 5 for an illustration. In Table 5, the ratio T_2/T_3 increases quickly as s increases, which can be justified theoretically; see the proof of Remark 2. Overall speaking, we should choose a UPD as the initial design in the LE procedure which will lead to a better search space. By Lemma 1, OAs reach the lower bound ϕ_{LB2} for ϕ values and satisfy $A_2 = 0$. As $T_3 \geqslant 0$, if the corresponding OAs are available, we should choose them as the initial designs.

Since the L_1-distance is equivalent to the Hamming distance in two-level designs, we can prove the following Corollary 2. It suggests that when starting from two-level initial designs, we should choose initial designs with small A_2 values in LE.

Corollary 2. *For any balanced $(n, 2^k)$ design D, when all possible level expansions are considered, we have*

(1) $\overline{\phi}_{\mathcal{E}}\left(D'\right) = \frac{((2m^2+1)n-6m^2)^2}{288k(k-1)m^4(n-2)^2} A_2(D) + C_3$, where C_3 is a constant;

(2) specifically, when D' is an LHD(n, k) $(m = n/s)$, $\overline{\phi}_{\mathcal{E}}\left(D'\right) = \frac{(n-1)^2}{72k(k-1)n^2} A_2(D) + C_4$, where C_4 is a constant.

Example 3. Consider the same 2^{10-3} and 3^{7-2} designs in Example 2, from which we randomly generate 10^5 LHDs via LE. We show their minimum ("Min" column) and sample average ("Sam.Ave" column) ϕ values in Tables 6 and 7, respectively. We also give the true average ("True.Ave" column) ϕ values of all possible generated designs in $\mathcal{E}(D)$ by Theorem 2.

From Tables 6 and 7, it is seen that the minimum and average ϕ values of the generated designs depend on the properties of the initial designs. Specifically, initial designs with smaller ϕ values will lead to better generated designs with smaller average (and minimum) ϕ values via LE, which illustrates Theorem 2. As a special case, two-level designs with smaller A_2 values will have smaller ϕ values, which illustrates Corollary 2. In addition, it is seen that the sample average is close to the true average in this example. Note that the true average ϕ values of all generated designs are calculated

Table 6
The ϕ-values (multiplied by 10^4) of designs generated from five 2^{10-3} initial designs via LE.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Sam.Ave	True.Ave
(0,0,0,3)	466.58	**1.40**	**1.76**	**1.76**
(0,0,1,2)	466.58	1.41	**1.76**	**1.76**
(0,1,0,2)	470.05	2.82	3.28	3.28
(0,2,0,1)	473.52	4.24	4.80	4.80
(0,3,0,3)	477.00	5.67	6.32	6.32

Table 7
The ϕ-values (multiplied by 10^4) of designs generated from five 3^{7-2} initial designs via LE.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Sam.Ave	True.Ave
(0,0,2,6)	199.76	0.53	**0.67**	**0.67**
(0,0,4,2)	199.76	0.51	**0.67**	**0.67**
(0,0,8,0)	199.76	**0.47**	**0.67**	**0.67**
(0,2,0,4)	206.29	4.58	4.89	4.89
(0,4,0,4)	212.82	8.70	9.12	9.12

by Theorem 2, and since it would be impossible to enumerate the $n_{\mathcal{E}}$ candidate designs for each case in Tables 6 and 7, we take 10^5 samples as an explanation.

3.3. Designs generated via BLPE

From an initial low-level design $D(n, s^k)$, we can perform both level permutation and expansion (BLPE) to generate high-level designs $D'(n, (ms)^k)$. Let $\Theta(D)$ represent the set of all designs generated via BLPE. Let $\overline{\phi}_{\Theta}\left(D'\right)$ be the average ϕ value of all designs in $\Theta(D)$.

Theorem 3. *From a balanced (n, s^k) initial design D, when all possible level permutations and expansions of D are considered, we have*

$$\overline{\phi}_{\Theta}\left(D'\right) = \frac{1}{n_{\Theta}} \sum_{D' \in \Theta(D)} \phi(D') = \frac{\left(m^2 s^2 - (n-1)m^2 s - n\right)^2}{18k(k-1)m^4 s^4 (n-s)^2} A_2(D) + C_5,$$

where the constant $n_{\Theta} = (s!)^k \left((n/s)!/(r!)^m\right)^{sk}$, $r = n/(ms)$, $m = n/s$,

$$C_5 = C_1 + C(k,s) - \frac{\left(s^2-1\right)\left((25k+3)s^2 - 25k - 7\right)}{480(k-1)s^4} - \frac{\left(s^2-1\right)^2}{288 s^4} - \frac{\left(s^2-1\right)\left[(2k+1)m^2 s^3 - (2k+1)n m^2 s^2 - 4kns - 2(k-1)n(m^2-2) + 2(2kn-k+1)m^2 s\right]}{72(k-1)m^2 s^4 (n-s)},$$

C_1 and $C(k, s)$ are defined in (3) and (1), respectively.

By Theorem 3, it is clear that from an initial design D with a smaller A_2 value, the designs generated via BLPE will have better average uniform projection property. Clearly, $\overline{\phi}_{\Theta}\left(D'\right)$ reaches the minimum when the initial design is an OA ($A_2(D) = 0$), which will lead to the best search space $\Theta(D)$.

Example 4. Consider the same 2^{10-3} and 3^{7-2} designs in Example 2, from which we randomly generate 10^5 LHDs via BLPE. We show their minimum and sample average ϕ values in Tables 8 and 9, respectively. We also give the true average ϕ values of all generated designs in $\Theta(D)$ by Theorem 3.

From Tables 8 and 9, it is clear that the minimum and average ϕ values of the generated designs depend on the properties of the initial designs. Specifically, the initial designs with smaller A_2 values will lead to better generated designs (i.e., smaller ϕ values) on average via BLPE, which illustrates Theorem 3.

3.4. Choices of OAs as initial designs

Based on the theoretical results in previous subsections, OAs ($A_2 = 0$) are good choices for initial designs if they are available. When LE is applied, various OAs may be available. For example, to generate a 16-run, 2-factor LHD, there are various 2-level and 4-level OAs that can be selected as initial designs in LE. In this subsection, we further detail the choices of OAs.

Table 8
The ϕ-values (multiplied by 10^4) of designs generated from five 2^{10-3} initial designs via BLPE.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Sam.Ave	True.Ave
(0,0,0,3)	466.58	1.43	**1.76**	**1.76**
(0,0,1,2)	466.58	**1.39**	**1.76**	**1.76**
(0,1,0,2)	470.05	2.85	3.28	3.28
(0,2,0,1)	473.52	4.27	4.80	4.80
(0,3,0,3)	477.00	5.70	6.32	6.32

Table 9
The ϕ-values (multiplied by 10^4) of designs generated from five 3^{7-2} initial designs via BLPE.

(A_1, A_2, A_3, A_4)	$\phi(D)$	$\phi(D')$		
		Min	Sam.Ave	True.Ave
(0,0,2,6)	199.76	0.53	**0.67**	**0.67**
(0,0,4,2)	199.76	0.51	**0.67**	**0.67**
(0,0,8,0)	199.76	**0.50**	**0.67**	**0.67**
(0,2,0,4)	206.29	2.77	3.59	3.59
(0,4,0,4)	212.82	5.01	6.50	6.50

Table 10
The average ϕ-values (multiplied by 10^4) and ϕ-efficiencies (%) of LHDs generated from OAs with different levels.

n	k	Lower levels			Higher levels		
		D_1	$\overline{\phi}(D_1')$	ϕ_{RE}	D_2	$\overline{\phi}(D_2')$	ϕ_{RE}
16	5	OA(16,5,2,2)	21.34	90.23	OA(16,5,4,2)	**16.28**	**93.70**
32	5	OA(32,5,2,2)	8.61	95.31	OA(32,5,4,2)	**5.98**	**97.11**
64	6	OA(64,6,2,3)	3.78	97.69	OA(64,6,4,3)	**2.44**	**98.61**
81	6	OA(81,6,3,2)	2.24	98.64	OA(81,6,9,2)	**1.07**	**99.45**
256	6	OA(256,6,4,2)	0.51	99.67	OA(256,6,16,2)	**0.17**	**99.90**

Given the run and factor sizes, different initial designs $OA(n, k, s, t)$ are characterized by the level sizes (s) and strengths (t). Note that an OA of strength t satisfies $A_1 = \cdots = A_t = 0$. Since only the A_2 value matters in Theorem 2, any strength $t \geqslant 2$ does not make a difference. Next, we study the influence of s on the average uniform projection properties of the generated LHDs in the LE procedure. By Theorem 2 and the properties of OAs, we can prove the following corollary.

Corollary 3. *For all possible LHDs $D'(n, k)$ generated from an $OA(n, k, s, t)$ via LE ($m = n/s$), we have*

$$\overline{\phi}_{\mathcal{E}}\left(D'\right) = \frac{(34n^2 - 4n - 5)s^2 + 8(n-1)n^2 s - 4n^3 + 4n^2}{144n^4 s^2} + \frac{1 + (-1)^n}{64n^4}. \quad (5)$$

Taking the derivative of $\overline{\phi}_{\mathcal{E}}\left(D'\right)$ in (5) with respect to s, we have

$$\frac{d\left(\overline{\phi}_{\mathcal{E}}\left(D'\right)\right)}{ds} = \frac{(n-1)\left(-s^2 + s\right)}{18n^2 s^4} < 0 \text{ for } s \geqslant 2.$$

Thus, $\overline{\phi}_{\mathcal{E}}\left(D'\right)$ decreases monotonically with respect to s. Given n and k, we should choose an OA with a large s so that the LHDs generated via LE will have better average uniform projection properties.

Example 5. Consider initial lower-level OAs D_1 and higher-level OAs D_2, from which we randomly generate 10^5 LHDs via LE. We show their average ϕ-values and ϕ-efficiencies in Table 10. It is seen that higher-level OAs D_2 will lead to better generated designs compared to lower-level OAs D_1 in terms of the average uniform projection properties.

4. Construction methods and numerical results

4.1. Construction guidelines

Based on the theoretical results in Section 3, we propose four construction methods for UPDs: (1) level permutation (LP), (2) level expansion (LE), (3) both level permutation and expansion (BLPE) and (4) step-by-step level permutation and expansion (SLPE). In this part, we first discuss how to choose the most appropriate method according to the required design sizes, and then illustrate the three-step procedure of the SLPE method.

Case 1: LP should be used when the required UPDs have prime numbers of levels s ($s > 2$). In such cases, if OAs are available, we can use them as UPDs according to Lemma 1. When OAs are not available, according to Theorem 1, we can select near OAs (Lu et al., 2006; Wang and Wu, 1992; Xu, 2002) or some random designs with small A_2 values as initial designs, and then perform LP to improve their uniform projection properties. Such initial designs will lead to the "average-best" search space having $(s!)^k$ generated designs. Note that LE cannot generate designs with prime numbers of levels, and thus BLPE and SLPE are equivalent to LP in this case.

Case 2: LE should be used when practitioners require high-level UPDs that can be generated from some low-level OAs. OAs are the desirable initial designs in LE, since they reach the lower bounds ϕ_{LB2} (for ϕ values) and satisfy $A_2 = 0$ by Lemma 1. According to Theorem 2 and Remarks 1 and 2, initial designs with small ϕ and A_2 values are preferred. In addition, by Corollary 3, an initial OA having as high the level s as possible should be chosen. Such an initial design will lead to the "average-best" search space with $((n/s)!/(r!)^m)^{sk}$ generated designs.

Case 3: BLPE should be used when the required UPDs are small with non-prime numbers of levels and they will be generated from non-OA initial designs. BLPE combines LP and LE, and will generally lead to better results compared to LE only. However, its search space includes as many as $(s!)^k\,((n/s)!/(r!)^m)^{sk}$ candidate designs, which can be very large for constructing large UPDs. By Theorem 3, we should select initial designs with small A_2 values (e.g. near OAs), which will lead to the "average-best" search space.

Case 4: SLPE should be used when the required UPDs have moderate or large sizes and non-prime numbers of levels. It has the following three steps.

(1) Given the required run size n and factor size k, choose an initial design with as small A_2 value as possible. Denote it by $D(n, s^k)$.
(2) If D is not an OA, perform LP on D and identify the best generated design with the smallest ϕ value. Denote it by $D_p(n, s^k)$.
(3) Starting from $D_p(n, s^k)$, perform LE to find the best generated UPD $D'(n, (ms)^k)$ with the smallest ϕ value.

SLPE combines LP and LE in a more efficient way compared to BLPE. The search space of SLPE includes $(s!)^k + ((n/s)!/(r!)^m)^{sk}$ designs, which is much smaller than that of BLPE. Thus, for moderate and large designs, SLPE will generally lead to better results compared to LE and BLPE. The efficiency of SLPE's first and second steps is proven by Theorem 1. The efficiency of SLPE's third step is proven by Theorem 2 and Remarks 1 and 2, where the average uniform projection property of all generated designs $\overline{\phi}_{\mathcal{E}}\left(D'\right)$ is dominated by the initial design's uniform projection property $\phi(D)$. Note that when OAs are available to be the initial designs in the first step, the second step of SLPE will be skipped and it reduces to the LE method.

In all of the above four cases, we aim to find the best search space of UPDs. Standard global optimization algorithms can be used to perform the search (Dueck and Scheuer, 1990; Morris and Mitchell, 1995; Kennedy and Eberhart, 1995; Holland, 1992). In this paper, we adopt the threshold accepting (TA) algorithm (Dueck and Scheuer, 1990; Xiao and Xu, 2018), which can be implemented with the R package "NMOF" (Schumann, 2021). We tailored this TA algorithm for LP, LE, BLPE, SLPE as well as searching for near OAs. Its pseudo code (Algorithm 1) and additional details are reported in Appendix A.

4.2. Numerical results

The design space for UPD grows exponentially fast as the design size increases, and a direct search over the entire space can become time-consuming and inefficient. The key idea in using level permutation and/or level expansion is to select efficient sub-spaces for searching. In this section, we compare our proposed methods to a direct search method via numerical studies. Additionally, we also compare the proposed SLPE to the LE and BLPE methods to justify the guidelines in Section 4.1.

In the current literature, researchers have adopted a direct search over the entire design space to identify UPDs (Sun et al., 2019). Here, we compare the proposed SLPE to the direct search method for generating uniform projection LHDs with various design sizes. In Table 11, we report the minimum, average and decrement rate of the generated LHDs' ϕ values. Here, the SLPE method starts from OAs or near OAs (marked with asterisks) and runs in less than five minutes for all cases. For the direct search method, we evaluate the ϕ values of randomly drawn LHDs, which takes longer computing time compared to the SLPE in every case. In Table 11, the decrement rate is a metric to evaluate the relative difference between the ϕ values of two designs. Specifically, for two designs D_1 and D_2 ($\phi(D_2) \leqslant \phi(D_1)$), the decrement rate is $\phi_{DR} = \left(\phi(D_1) - \phi(D_2)\right)/\phi(D_1) \times 100\%$. Clearly, $0 \le \phi_{DR} \le 1$, and large ϕ_{DR} means design D_2 is much better than design D_1 in terms of the uniform projection criterion.

From Table 11, it is seen that the SLPE is superior compared to the direct search in terms of both the minimum and average ϕ values in all cases. In addition, as the design size increases, the decrement rate of the ϕ values of the SLPE relative to the direct search increases rapidly, thus the advantage of SLPE becomes more obvious.

Next, we compare the SLPE method to the BLPE method. As discussed in the guidelines, the search space of BLPE also grows fast as the design size increases, and the SLPE chooses an efficient sub-space of it. In Table 12, we report the minimum, average and decrement rate of the ϕ values for the generated LHDs from both methods. For the initial designs, we consider two 3^{7-2} designs with different GWLPs (the first two cases in Table 12) and several near OAs with small A_2 values. For all cases, we let the BLPE run for longer time than the SLPE, which takes several seconds to several minutes (varying by cases). From Table 12, it is seen that the SLPE outperforms the BLPE for moderate and large design sizes. Note

Table 11
Comparison of uniform projection LHDs constructed by the direct search and SLPE methods.

Initial design	Direct search		SLPE		Decrement rate	
	ϕ_{min}	$\bar{\phi}$	ϕ_{min}	$\bar{\phi}$	Min (%)	Ave (%)
$(15, 3^3)^*$	16.25	27.69	**14.12**	**15.25**	13.11	44.93
$(21, 3^4)^*$	11.11	17.92	**7.79**	**8.76**	29.84	51.13
$(25, 5^5)$	9.12	14.43	**5.80**	**6.08**	36.43	57.83
$(27, 3^3)$	6.62	13.14	**4.60**	**5.33**	30.47	59.47
$(30, 5^5)^*$	7.32	11.56	**4.20**	**5.02**	42.62	56.60
$(40, 2^4)$	4.39	8.24	**2.37**	**3.33**	45.93	59.52
$(49, 7^3)$	3.15	6.52	**1.49**	**1.70**	52.81	73.97
$(50, 5^5)$	3.73	6.40	**1.58**	**1.86**	57.49	70.96
$(64, 8^4)$	2.57	4.85	**0.95**	**1.09**	63.12	77.62
$(64, 4^{20})$	4.28	4.84	**1.62**	**1.80**	62.24	62.92
$(75, 5^5)^*$	2.52	4.06	**0.80**	**1.00**	68.24	75.38
$(81, 9^8)$	2.79	3.75	**0.71**	**0.78**	74.55	79.12
$(128, 8^{12})$	1.84	2.30	**0.38**	**0.43**	79.52	81.12

Table 12
Comparison of the BLPE and the SLPE methods.

Initial design	BLPE		SLPE		Decrement rate	
	ϕ_{min}	$\bar{\phi}$	ϕ_{min}	$\bar{\phi}$	Min (%)	Ave (%)
$(243, 3^7)_1$	2.79	3.59	**2.34**	**2.46**	16.13	31.48
$(243, 3^7)_2$	5.02	6.49	**4.49**	**4.61**	10.56	28.97
$(15, 3^3)^*$	16.03	22.16	**14.12**	**15.25**	11.94	31.19
$(21, 3^4)^*$	10.14	13.47	**7.79**	**8.76**	23.12	34.97
$(35, 7^7)^*$	5.39	6.73	**3.53**	**3.71**	34.53	44.90
$(45, 9^5)^*$	3.48	5.06	**2.13**	**2.26**	38.78	55.23
$(70, 10^7)^*$	2.10	2.75	**1.05**	**1.15**	50.10	58.29
$(75, 15^5)^*$	1.98	3.30	**0.92**	**0.98**	53.39	70.39

Note: $(243, 3^7)_1$ refers to the 3^{7-2} design with $(A_1, A_2, A_3, A_4) = (0, 2, 0, 4)$, and $(243, 3^7)_2$ refers to the 3^{7-2} design with $(0, 4, 0, 4)$.

Table 13
Comparisons of the LE and the SLPE methods.

Initial design	LE		SLPE		Decrement rate	
	ϕ_{min}	$\bar{\phi}$	ϕ_{min}	$\bar{\phi}$	Min (%)	Ave (%)
$(243, 3^7)_1$	4.63	4.89	**2.34**	**2.46**	49.46	49.69
$(243, 3^7)_2$	8.76	9.12	**4.49**	**4.61**	48.74	49.45
$(30, 6^3)^*$	4.89	6.66	**3.85**	**4.16**	21.19	37.45
$(35, 7^7)^*$	5.89	6.64	**3.53**	**3.71**	40.07	44.12
$(45, 9^5)^*$	3.77	4.37	**2.13**	**2.26**	43.61	48.20
$(70, 10^7)^*$	2.26	2.52	**1.05**	**1.15**	53.62	54.51
$(75, 15^5)^*$	3.21	3.50	**0.92**	**0.98**	71.26	72.13

Note: $(243, 3^7)_1$ refers to the 3^{7-2} design with $(A_1, A_2, A_3, A_4) = (0, 2, 0, 4)$, and $(243, 3^7)_2$ refers to the 3^{7-2} design with $(0, 4, 0, 4)$.

that the BLPE method is superior to the direct search method. As the design size increases, the advantage of SLPE becomes more obvious since the decrement rate increases.

Finally, we compare the SLPE method to the LE method. As illustrated in Section 4.1, the LE is the third step of the SLPE. When OAs are available as the initial designs, they are essentially the same, since the first two steps of the SLPE should be skipped. Here, we consider some cases where OAs are not available. In Table 13, we report the minimum, average and decrement rate of the ϕ values for the LHDs generated by the LE and the SLPE methods. In all cases, we let the LE run for longer time than SLPE, which takes several seconds to several minutes (varying by cases). From Table 13, it is seen that the SLPE is superior to the LE for all cases where OAs are not available as initial designs.

5. Discussion

When only portions of the input factors are active in computer experiments, the low-dimensional projection uniformity of designs is important. UPDs focus on the uniformity over two-dimensional projections and also have good space-filling properties over all projections. In this paper, we propose to construct UPDs with flexible sizes via (1) level permutation (LP), (2) level expansion (LE), (3) both level permutation and expansion (BLPE), and (4) step-by-step level permutation and expansion (SLPE). Theoretical results are developed to connect the uniform projection properties of the generated designs to the properties of the initial designs in the proposed methods, and they will guide the search algorithm focusing on efficient sub-spaces of solutions. Guidelines are provided for choosing appropriate construction methods according to the required design sizes, and numerical results are presented to illustrate the efficiency of the proposed methods.

Although the SLPE can effectively generate uniform projection designs with flexible sizes, its LE step may be further improved when starting from OAs ($A_2 = 0$). We find that the optimal UPD is often a mirror-symmetric design (Tang and Xu, 2014). Thus, certain structural level expansion may lead to OA-based LHDs with good uniform projection properties. This can be an interesting topic for future research. Moreover, Sun et al. (2019) established a link between the uniform projection criterion and the L_1-distance of a design, and Wang et al. (2020) established the relationship between the orthogonality criterion and the L_2-distance of a design. Their corresponding formulations are very similar, but the ranges of variation of the ϕ value and the orthogonality metric value are not the same (which depend on the design sizes). From this consideration, we can propose a new criterion that can be used to evaluate designs' uniform projection properties and orthogonality in a more comprehensive way as a future research.

Acknowledgments

The research was supported by National Natural Science Foundation of China (11971098, 11471069) and the National Key Research and Development Program of China (2020YFA0714102).

Appendix A. Technical details on ta optimization

The threshold accepting (TA) algorithm is a widely used global optimization algorithm (Dueck and Scheuer, 1990; Xiao and Xu, 2018). To avoid falling into a local optimal solution, the TA algorithm will accept a new solution that is not much worse than the old one. We briefly describe its work-flow for identifying optimal designs here. Starting from an initial solution (design), the current solution is changed through its neighbor in each iteration, and the new solution is accepted if its objective function value improves or worsens less than a threshold. The threshold values are generated by the empirical distributions of increments for the object function. As the iteration increases, the threshold values decrease and the search tends to become more stable with less "jumps". The pseudo code for the TA algorithm is reported in the following Algorithm 1.

Algorithm 1 A TA algorithm

Initialize tuning parameters n_{seq} (number of iterations to compute the threshold sequence), n_{rounds} (number of rounds) and n_{steps} (number of steps).
Initialize a starting design D_0; set $D_{opt} = D_c = D_0$.
for $j = 1$ to n_{seq} **do**
    Generate a neighbor solution $N(D_c)$ and let $\Delta_j = |f(D_c) - f(N(D_c))|$.
end for
Compute the empirical distribution of $\Delta_j, j = 1, 2, \dots, n_{seq}$, denoted it as F.
for $r = 1$ to n_{rounds} **do**
    Generate thresholds $\tau_r = F^{-1}(0.5(1 - r/n_{rounds}))$.
    for $i = 1$ to n_{steps} **do**
        Generate a neighbor solution $N(D_c)$ and let $\delta = f(N(D_c)) - f(D_c)$.
        if $\delta < \tau_r$, **then** let $D_c = N(D_c)$.
        if $f(D_c) < f(D_{opt})$, **then** let $D_{opt} = D_c$.
    end for
end for
Return D_{opt}.

This TA algorithm will be used in both LP and LE as well as for searching near OAs. Note that the BLPE and the SLPE are combinations of the LP and the LE methods. Specifically, for the LP procedure, the neighbor design $\mathcal{N}(D_c)$ in the TA algorithm is obtained by exchanging all elements of two random levels in a randomly chosen column of the current design D_c. For the LE procedure, the neighbor design $\mathcal{N}(D_c)$ is obtained by exchanging the levels in two positions from a randomly chosen column of D_c, where these two positions have different values in D_c but the same value in its corresponding initial design. The criterion ϕ in (1) is used as the objective function f here. When searching for near OAs, the neighbor design $\mathcal{N}(D_c)$ is obtained by exchanging two random positions in a randomly chosen column of the current design D_c. The criterion A_2 is used as the objective function f. In Algorithm 1, we typically set the tuning parameters n_{seq} from 500 to 2000, n_{rounds} from 10 to 50 and n_{steps} from 1000 to 5000 according to the practical needs.

Appendix B. Proofs

We first present a lemma that will be used in the subsequent proofs.

Lemma 3. *For a balanced (n, s^k) design $X = (x_{il})_{n\times k}$ with levels from $\{1, \dots, s\}$, suppose $h_{i,j}$, $d_{i,j}$ and $d_2(x_i, x_j)$ are Hamming distance, L_1- and L_2-distance of two rows $x_i = (x_{i1}, \dots, x_{ik})$ and $x_j = (x_{j1}, \dots, x_{jk})$ respectively. We have*

$$\sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j} = \frac{kn^2(s-1)}{s}, \tag{B.1}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j}^2 = \frac{n^2}{s^2}\left\{2A_2(D) + (s-1)k[1+(s-1)k]\right\}, \tag{B.2}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j} = \frac{kn^2\left(s^2-1\right)}{3s}, \tag{B.3}$$

$$\sum_{i=1}^{n} d_2^2\left(x_i, s_0\right) = \frac{kn\left(s^2-1\right)}{12}, \tag{B.4}$$

$$\sum_{i=1}^{n} d_4^4\left(x_i, s_0\right) = \frac{kn\left(s^2-1\right)\left(3s^2-7\right)}{240}, \tag{B.5}$$

where $s_0 = (s+1)/2$, $d_2(x_i, s_0) = \left(\sum_{l=1}^{k}|x_{il} - s_0|^2\right)^{1/2}$ and $d_4(x_i, s_0) = \left(\sum_{l=1}^{k}|x_{il} - s_0|^4\right)^{1/4}$. Eqs. (B.1) and *(B.2) refer to Xu (2003), which demonstrates the relationship between the generalized word-length pattern and Hamming distances. The remaining equations in Lemma 3 can be proven via tedious calculations, so we omit the details.*

Proof of Theorem 1. We prove this result by using Theorem 3.1 of Tang and Xu (2013) and induction.

For any U-type design D with 1 or 2 columns, the result follows directly by Theorem 3.1 of Tang and Xu (2013). Suppose that the result holds for any U-type (n, s^{k-1}) design with $k-1 \geq 3$. For any U-type (n, s^k) design D, we partition D as $(D_{\{1\}}, D_{\{2,\dots,k\}})$. Then,

$$\begin{aligned}
\overline{\phi}_{\mathcal{P}}\left(D'\right) &= \frac{1}{(s!)^k}\sum_{D^*_{\{1\}}\in\mathcal{P}(D_{\{1\}})}\ \sum_{D^*_{\{2,\dots,k\}}\in\mathcal{P}(D_{\{2,\dots,k\}})} \phi\left(\left(D^*_{\{1\}}, D^*_{\{2,\dots,k\}}\right)\right)\\
&= \frac{1}{(s!)^k}\sum_{D^*_{\{1\}}\in\mathcal{P}(D_{\{1\}})}\ \sum_{D^*_{\{2,\dots,k\}}\in\mathcal{P}(D_{\{2,\dots,k\}})} \frac{1}{\binom{k}{2}}\left(\sum_{i=2,\dots,k} CD\left(D^*_{\{1,i\}}\right) + \sum_{|u|=2, u\subset\{2,\dots,k\}} CD\left(D^*_u\right)\right)\\
&= \frac{1}{\binom{k}{2}}\left(\sum_{i=2}^{k}\frac{(s+1)^2}{36s^4}A_2\left(D_{\{1,i\}}\right) + (k-1)\phi_{LB2}\right) + \frac{\binom{k-1}{2}}{\binom{k}{2}}\left(\frac{(s+1)^2}{18(k-1)(k-2)s^4}A_2\left(D_{\{2,\dots,k\}}\right) + \phi_{LB2}\right)\\
&= \frac{(s+1)^2}{18k(k-1)s^4}\left(A_2\left(D_{\{2,\dots,k\}}\right) + \sum_{i=2}^{k} A_2\left(D_{\{1,i\}}\right)\right) + \phi_{LB2},
\end{aligned}$$

where the second to last equation follows by induction. By the definition of $A_2(D)$, we have $A_2(D) = A_2(D_{\{2,\dots,k\}}) + \sum_{i=2}^{k} A_2(D_{\{1,i\}})$, which completes the proof.

To prove Theorem 2, we first prove Lemma 2.

Proof of Lemma 2. For a balanced (n, s^k) design $D = (x_{il})_{n\times k}$, combining (B.4) and (B.5), we can simplify $g(D)$ using Lemma 2 of Sun et al. (2019):

$$\begin{aligned}
g(D) &= \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - \frac{2}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{n} d_{i,j}\right)^2\\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - \frac{4n}{s^2}\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip} - s_0\right)^2\left(x_{iq} - s_0\right)^2 + C_0,
\end{aligned}$$

where $C_0 = -kn^2\left(s^2-1\right)\left((25k+3)s^2 - 25k - 7\right)/120s^2$. Thus, based on (1), we have

$$\phi(D) = \frac{1}{4k(k-1)n^2s^2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - \frac{1}{k(k-1)ns^4}\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip} - s_0\right)^2\left(x_{iq} - s_0\right)^2 + C_2,$$

where $C_2 = C(k, s) - \left(s^2-1\right)\left((25k+3)s^2 - 25k - 7\right)/480(k-1)s^4$ and $C(k, s)$ is defined in (1).

Proof of Theorem 2. Generate a series of $\left(n, (ms)^k\right)$ designs $D' = (x'_{il})_{n\times k}$ from an initial $\left(n, s^k\right)$ design $D = (x_{il})_{n\times k}$ via level expansion. First of all, we need to calculate the expectations of the first two terms in Eq. (4) by Lemma 2. Note that the generated designs take ms levels, so the s in (4) should be replaced by ms.

For the expectation of the first term, we first prove the result below based on Theorem 1 of Xiao and Xu (2018). For generated designs D', the second moment of their pairwise L_1-distance $d'_{i,j}$, for $i, j = 1, \ldots, n$ and $i \neq j$, has the following relationship with the initial design D:

$$E_{\mathcal{E}}\left(d'_{i,j}\right)^2 = m^2 d_{i,j}^2 + 2km\gamma d_{i,j} - 2m\gamma h_{i,j} d_{i,j} + \gamma^2 h_{i,j}^2 + \left(C_{2,1} - 2k\gamma^2\right) h_{i,j} + \left(C_{2,0} + k^2\gamma^2\right),$$

where $\gamma = n\left(m^2-1\right)/3m(n-s)$, $C_{2,0} = kn\left(m^2-1\right)\left(m^2 n + 2n - 3m^2 s\right)/18m^2(n-s)^2$ and $C_{2,1} = \left(m^2-1\right)\left[2n^2\left(m^2-1\right) - 3m^2 s(n-s)\right]/18m^2(n-s)^2$. Then, combining (B.1)–(B.3), we can obtain

$$E_{\mathcal{E}}\left(\sum_{i=1}^{n}\sum_{j=1}^{n} d'^2_{i,j}\right) = m^2 \sum_{i=1}^{n}\sum_{j=1}^{n} d_{i,j}^2 - \frac{2n\left(m^2-1\right)}{3(n-s)} \sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j} d_{i,j} + \frac{2n^4\left(m^2-1\right)^2}{9m^2 s^2 (n-s)^2} A_2(D) + C_{2,2}, \tag{B.6}$$

where the constant $C_{2,2} = k(k-1)n^3\left(m^2-1\right)\left(s^2 + n\left(m^2-1\right)\right)/9m^2 s^2 (n-s)^2 + kn^2\left(m^2-1\right)\left[-(4kn-3)s^3 + 2(2kn-3)ns^2 + (2k+3n+2)ns - 4kn^2\right]/18s(n-s)^2$.

For the expectation of the second term, we first prove the below conclusion:

$$\begin{aligned} E_{\mathcal{E}}\left(x'_{ip} - s'_0\right)^2 &= \sum_{t=1}^{m} \left(\left(x_{ip}-1\right)m + t - s'_0\right)^2 P\left(x'_{ip} = \left(x_{ip}-1\right)m + t\right) \\ &= \frac{1}{m}\sum_{t=1}^{m}\left(t + \left(\left(x_{ip}-1\right)m - s'_0\right)\right)^2 \\ &= m^2\left(x_{ip} - s_0\right)^2 + \frac{m^2-1}{12}, \end{aligned} \tag{B.7}$$

where $s'_0 = (ms+1)/2$. Since x'_{ip} and x'_{iq} $(p \neq q)$ are determined independently by the pth and qth columns in the initial design D, combining (B.4) and (B.7), we obtain

$$\begin{aligned} & E_{\mathcal{E}}\left(\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x'_{ip} - s'_0\right)^2\left(x'_{iq} - s'_0\right)^2\right) \\ &= \sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k} E_{\mathcal{E}}\left(x'_{ip} - s'_0\right)^2 E_{\mathcal{E}}\left(x'_{iq} - s'_0\right)^2 \\ &= m^4 \sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip} - s_0\right)^2\left(x_{iq} - s_0\right)^2 + \frac{nk(k-1)\left(m^2-1\right)\left(2m^2 s^2 - m^2 - 1\right)}{288}. \end{aligned} \tag{B.8}$$

Ultimately, because each level expansion occurs with equal probability in the sub-space $\mathcal{E}(D)$, combining (B.6), (B.8) and (4) we have

$$\begin{aligned} \overline{\phi}_{\mathcal{E}}\left(D'\right) = E_{\mathcal{E}}\left[\phi\left(D'\right)\right] &= \phi(D) - \frac{m^2-1}{6k(k-1)m^2 s^2 n(n-s)} \sum_{i=1}^{n}\sum_{j=1}^{n} h_{i,j} d_{i,j} \\ &\quad + \frac{n^2\left(m^2-1\right)^2}{18k(k-1)m^4 s^4 (n-s)^2} A_2(D) + C_1, \end{aligned}$$

where the constant

$$\begin{aligned} C_1 &= \frac{\left(m^2-1\right)\left[-(4kn-3)s^3 + 2(2kn-3)ns^2 + (2k+3n+2)ns - 4kn^2\right]}{72(k-1)m^2 s^3 (n-s)^2} \\ &\quad + \frac{n\left(m^2-1\right)\left(s^2 + n\left(m^2-1\right)\right)}{36m^4 s^4 (n-s)^2} - \frac{1+(-1)^s}{64s^4} - \frac{\left(m^2-1\right)\left(2m^2 s^2 - m^2 - 1\right)}{288m^4 s^4} \\ &\quad - \frac{\left(m^2-1\right)\left(2(11k-9)m^2 s^2 - 3(k-1)m^2 - 3k + 3\right)}{96(k-1)m^4 s^4} + \frac{1+(-1)^{ms}}{64m^4 s^4}. \end{aligned}$$

Proof of Remark 1. For a balanced (n, s^k) design $D = (x_{il})_{n\times k}$, to justify that the first term of $\phi(D)$ in (4) is greater than the absolute value of the second term, we can prove the following inequation:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\sum_{l=1}^{k}\left|x_{il}-x_{jl}\right|\right)^2 > \frac{4n}{s^2}\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip}-s_0\right)^2\left(x_{iq}-s_0\right)^2. \tag{B.9}$$

On the one hand, using (B.3) and the Cauchy–Schwarz inequality twice on the left side of (B.9), we can obtain

$$\begin{aligned}&\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\sum_{l=1}^{k}\left|x_{il}-x_{jl}\right|\right)^2 \geqslant \sum_{i=1}^{n}\left(\frac{1}{n}\left(\sum_{j=1}^{n}\sum_{l=1}^{k}\left|x_{il}-x_{jl}\right|\right)^2\right)\\ &\geqslant \frac{1}{n^2}\left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{l=1}^{k}\left|x_{il}-x_{jl}\right|\right)^2 = \frac{k^2n^2\left(s^2-1\right)^2}{9s^2}.\end{aligned} \tag{B.10}$$

On the other hand, using (B.5) and the Cauchy–Schwarz inequality once on the right side of (B.9), we can obtain

$$\begin{aligned}&\frac{4n}{s^2}\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip}-s_0\right)^2\left(x_{iq}-s_0\right)^2\\ &\leqslant \frac{4n}{s^2}\sum_{1\leqslant p<q\leqslant k}\left(\sum_{i=1}^{n}\left(x_{ip}-s_0\right)^4\right)^{\frac{1}{2}}\left(\sum_{i=1}^{n}\left(x_{iq}-s_0\right)^4\right)^{\frac{1}{2}} = \frac{n^2k(k-1)\left(s^2-1\right)\left(3s^2-7\right)}{120s^2}.\end{aligned} \tag{B.11}$$

In addition, since $k \geq 2$ and $s \geq 2$, it is easy to verify that

$$\frac{\frac{k^2n^2\left(s^2-1\right)^2}{9s^2}}{\frac{n^2k(k-1)\left(s^2-1\right)\left(3s^2-7\right)}{120s^2}} = \frac{40}{3}\left(1+\frac{1}{k-1}\right)\left(\frac{s^2-1}{3s^2-7}\right) \geqslant \frac{40}{9} > 1. \tag{B.12}$$

Thus, combining (B.10)–(B.12), we can prove (B.9).

Proof of Remark 2. For a balanced (n, s^k) design D, when all possible level expansions of D are considered, we find the following:

(1) For the coefficient part, since $n \geq ms$ and $m \geq 2$,

$$\frac{\frac{1}{4k(k-1)n^2s^2}}{\frac{m^2-1}{6k(k-1)m^2s^2n(n-s)}} = \frac{3m^2}{2\left(m^2-1\right)}\left(1-\frac{s}{n}\right) = \frac{3}{2}\left(1-\frac{1}{m+1}\right) \geqslant 1. \tag{B.13}$$

For the body part, since $(d_{i,j}-h_{i,j}) \geqslant 0$, we can consider $(d_{i,j}-h_{i,j})$ to be the L_1-distance between the ith row and the jth row of a new design. We define $\left(d_{i,j}-h_{i,j}\right) = \sum_{l=1}^{k} f_1\left(x_{il}, x_{jl}\right)$, where $f_1\left(x_{il}, x_{jl}\right)$ is equal to $\left|x_{il}-x_{jl}-1\right|$ if $x_{il} \neq x_{jl}$ and 0 otherwise. Furthermore, we define the L_2-distance between the ith row and the jth row of the new design as $\sqrt{\sum_{l=1}^{k} f_2\left(x_{il}, x_{jl}\right)}$, where $f_2\left(x_{il}, x_{jl}\right)$ is equal to $\left(x_{il}-x_{jl}-1\right)^2$ if $x_{il} \neq x_{jl}$ and 0 otherwise. By the norm inequality and several simple algebraic steps, we have

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\left(d_{i,j}-h_{i,j}\right)^2=\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\sum_{l=1}^{k}f_1\left(x_{il},x_{jl}\right)\right)^2 \geqslant \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{l=1}^{k}f_2\left(x_{il},x_{jl}\right)=\frac{kn^2(s-2)(s-1)^2}{6s}. \tag{B.14}$$

Since $h_{i,j}\leqslant k$, combining (B.3) and (B.14) yields

$$\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}d_{i,j}^2}{\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}}=1+\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(d_{i,j}-h_{i,j}\right)d_{i,j}}{\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}}\geqslant 1+\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(d_{i,j}-h_{i,j}\right)^2}{\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}}\geqslant 1+\frac{(s-2)(s-1)}{2k(s+1)}. \tag{B.15}$$

Therefore, combining (B.13) and (B.15), we obtain

$$\frac{T_1}{T_2}=\frac{\frac{1}{4k(k-1)n^2s^2}\sum_{i=1}^{n}\sum_{j=1}^{n}d_{i,j}^2}{\frac{m^2-1}{6k(k-1)m^2s^2n(n-s)}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}}\geqslant 1+\frac{(s-2)(s-1)}{2k(s+1)}.$$

(2) If the ratio of the second term to the third term in Eq. (3) is greater than or equal to 1, it is easy to obtain

$$A_2(D)\leqslant\frac{3m^2s^2(n-s)}{n^3(m^2-1)}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}.$$

For the lower bound of $(3m^2s^2(n-s))(n^3(m^2-1))^{-1}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}$, since $n\geq ms$, $m\geq 2$, $s\geq 2$ and $h_{i,j}\geqslant 1$, combining (B.3), we obtain

$$\frac{3m^2s^2(n-s)}{n^3\left(m^2-1\right)}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}\geqslant\frac{km^2s\left(s^2-1\right)}{m^2-1}\left(1-\frac{s}{n}\right)\geqslant\frac{kms\left(s^2-1\right)}{m+1}\geqslant 4k.$$

Proof of Corollary 2. For a design $D(n,2^k)$, its L_1-distances equal Hamming distances. (1) By Theorem 2, Corollary 1 and (B.2), we obtain

$$\overline{\phi}_{\mathcal{E}}\left(D'\right)=\frac{((2m^2+1)n-6m^2)^2}{288k(k-1)m^4(n-2)^2}A_2(D)+C_3,$$

where

$$C_3=\frac{(k+1)\left(\left(m^2+2\right)n-6m^2\right)}{192(k-1)m^2(n-2)}+\frac{n\left(m^2-1\right)\left(\left(m^2-1\right)n+4\right)}{576m^4(n-2)^2}-\frac{\left(4m^2-1\right)\left((20k+1)m^2-5k-1\right)}{1440(k-1)m^4}$$
$$+\frac{\left(m^2-1\right)\left((6k+3)n^2-(14k+10)n+12\right)}{288(k-1)m^2(n-2)^2}+\frac{1}{512m^4}+\frac{64(5k-2)m^4+120(3k-5)m^2+15k+33}{11520(k-1)m^4}.$$

(2) Specifically, when the generated design D' is an $LHD(n,k)$ $(m=n/s)$, we can simplify this further:

$$\overline{\phi}_{\mathcal{E}}\left(D'\right)=\frac{(n-1)^2}{72k(k-1)n^2}A_2(D)+C_4,$$

where $C_4=(6n^3+62n^2-8n-1)/(288n^4)$.

Proof of Theorem 3. Let σ represent a level permutation procedure, π represent a level expansion procedure, and Θ represent the set of all designs generated via all possible level permutations and expansions. Let E_σ express the expectation of the designs generated via all possible level permutations, E_π express the expectation of the designs generated via all possible level expansions and E_Θ express the expectation of the designs generated via all possible level permutations and expansions. Using the properties of conditional expectations, we obtain

$$E_\Theta\left[\phi(D')\right]=E_\sigma\left[E_\pi\left(\phi(D')|\sigma\right)\right]. \tag{B.16}$$

Given a level permutation procedure σ, let $d_{i,j}^{\sigma}$ represent the L_1-distance of a design generated via σ. The level permutation does not change the pairwise Hamming distances or A_2 value of a design. Using Theorem 2, we obtain

$$E_\pi\left[\phi\left(D'\right)|\sigma\right]=(\phi(D))^{\sigma}-\frac{m^2-1}{6k(k-1)m^2s^2n(n-s)}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}^{\sigma}+\frac{n^2\left(m^2-1\right)^2}{18k(k-1)m^4s^4(n-s)^2}A_2(D)+C_1$$
$$=\frac{1}{4k(k-1)n^2s^2}\sum_{i=1}^{n}\sum_{i=1}^{n}\left(d_{i,j}^{\sigma}\right)^2+\frac{n^2\left(m^2-1\right)^2}{18k(k-1)m^4s^4(n-s)^2}A_2(D)+C_{3,1}$$

$$-\frac{1}{k(k-1)ns^4}\sum_{i=1}^{n}\sum_{1\leqslant p<q\leqslant k}\left(x_{ip}^{\sigma}-s_0\right)^2\left(x_{iq}^{\sigma}-s_0\right)^2-\frac{m^2-1}{6k(k-1)m^2s^2n(n-s)}\sum_{i=1}^{n}\sum_{j=1}^{n}h_{i,j}d_{i,j}^{\sigma}, \tag{B.17}$$

where $C_{3,1}=C_1+C(k,s)-\left(s^2-1\right)\left((25k+3)s^2-25k-7\right)/480(k-1)s^4$.

Xiao and Xu (2018) pointed out that the expectation and variance of $d_{i,j}^{\sigma}$ have the following relationships: $E_{\sigma}\left(d_{i,j}^{\sigma}\right)=(s+1)h_{i,j}/3$ and $\mathrm{Var}_{\sigma}\left(d_{i,j}^{\sigma}\right)=(s+1)(s-2)h_{i,j}/18$. Thus, we can obtain the second moment of $d_{i,j}^{\sigma}$:

$$E_{\sigma}\left(d_{i,j}^{\sigma}\right)^2=\frac{(s+1)(s-2)}{18}h_{i,j}+\frac{(s+1)^2}{9}h_{i,j}^2. \tag{B.18}$$

In addition, we have

$$E_{\sigma}\left(x_{ip}^{\sigma}-s_0\right)^2=\sum_{t=1}^{s}(t-s_0)^2P\left(x_{ip}^{\sigma}=t\right)=\frac{1}{s}\sum_{t=1}^{s}(t-s_0)^2=\frac{s^2-1}{12}. \tag{B.19}$$

Since the pth and qth columns of the generated design are determined independently in the level permutation procedure, combining Theorem 2, (B.1), (B.2) and (B.16)–(B.19), after performing some simple algebra, we obtain

$$\overline{\phi}_{\Theta}\left(D'\right)=E_{\Theta}\left[\phi\left(D'\right)\right]=\frac{\left(m^2s^2-(n-1)m^2s-n\right)^2}{18k(k-1)m^4s^4(n-s)^2}A_2(D)+C_5,$$

where the constant

$$C_5=C_1+C(k,s)-\frac{\left(s^2-1\right)\left((25k+3)s^2-25k-7\right)}{480(k-1)s^4}-\frac{\left(s^2-1\right)^2}{288s^4}$$
$$-\frac{\left(s^2-1\right)\left[(2k+1)m^2s^3-(2k+1)nm^2s^2-4kns-2(k-1)n(m^2-2)+2(2kn-k+1)m^2s\right]}{72(k-1)m^2s^4(n-s)},$$

and C_1 is defined in (3).

Appendix C. Codes

For the codes used for the SLPE, please see https://github.com/Yishan130426/TA_UPD.

References

Dueck, G., Scheuer, T., 1990. Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. J. Comput. Phys. 90, 161–175.
Fang, K.T., Li, R., Sudjianto, A., 2006. Design and Modeling for Computer Experiments. Chapman & Hall/CRC, New York.
Garud, S.S., Karimi, I.A., Kraft, M., 2017. Design of computer experiments: A review. Comput. Chem. Eng. 106, 71–95.
Gramacy, R.B., 2020. Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences. Chapman & Hall/CRC, New York.
Hedayat, A.S., Sloane, N.J., Stufken, J., 1999. Orthogonal Arrays: Theory and Applications. Springer, New York.
Hickernell, F.J., 1998. A generalized discrepancy and quadrature error bound. Math. Comp. 67, 299–322.
Holland, J.H., 1992. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. MIT Press, Cambridge.
Jiang, B.C., Ai, M.Y., 2017. Construction of uniform U-designs. J. Statist. Plan. Inference 181, 1–10.
Johnson, M.E., Moore, L.M., Ylvisaker, D., 1990. Minimax and maximin distance designs. J. Statist. Plan. Inference 26, 131–148.
Joseph, V.R., 2016. Space-filling designs for computer experiments: A review. Qual. Eng. 28, 28–35.
Joseph, V.R., Gul, E., Ba, S., 2015. Maximum projection designs for computer experiments. Biometrika 102, 371–380.
Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4. IEEE, pp. 1942–1948.
Kleijnen, J.P., 2017. Design and analysis of simulation experiments: Tutorial. In: Tolk, A., Fowler, J., Shao, G., Yucesan, E. (Eds.), Advances in Modeling and Simulation. Springer, New York, pp. 135–158.
Leary, S., Bhaskar, A., Keane, A., 2003. Optimal orthogonal-array-based latin hypercubes. J. Appl. Stat. 30, 585–598.
Li, W.L., Liu, M.Q., Tang, B.X., 2020. A method of constructing maximin distance designs. Biometrika http://dx.doi.org/10.1093/biomet/asaa089.
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 & Hall/CRC, New York, pp. 593–625.
Lu, X., Li, W., Xie, M., 2006. A class of nearly orthogonal arrays. J. Qual. Technol. 38, 148–161.
Lukemire, J., Xiao, Q., Mandal, A., Wong, W.K., 2021. Statistical analysis of complex computer models in astronomy. arXiv preprint arXiv:2102.07179.
McKay, M.D., Beckman, R.J., Conover, W.J., 1979. A 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.M., Santner, T.J., 2012. Two-stage sensitivity-based group screening in computer experiments. Technometrics 54, 376–387.
Morris, M., Mitchell, T.J., 1995. Exploratory designs for computational experiments. J. Statist. Plan. Inference 43, 381–402.
Santner, T.J., Williams, B.J., Notz, W.I., 2003. The Design and Analysis of Computer Experiments. Springer, New York.
Schumann, E., 2021. Numerical methods and optimization in finance (nmof) manual. package version 2.4-1. http://enricoschumann.net/NMOF/.
Sun, F.S., Tang, B.X., 2017a. A general rotation method for orthogonal latin hypercubes. Biometrika 104, 465–472.
Sun, F.S., Tang, B.X., 2017b. A method of constructing space-filling orthogonal designs. J. Amer. Statist. Assoc. 112, 683–689.
Sun, F.S., Wang, Y.P., Xu, H.Q., 2019. Uniform projection designs. Ann. Statist. 47, 641–661.
Tang, B.X., 1993. Orthogonal array-based latin hypercubes. J. Amer. Statist. Assoc. 88, 1392–1397.

Tang, Y., Xu, H.Q., 2013. An effective construction method for multi-level uniform designs. J. Statist. Plan. Inference 143, 1583–1589.
Tang, Y., Xu, H.Q., 2014. Permuting regular fractional factorial designs for screening quantitative factors. Biometrika 101, 333–350.
Tang, Y., Xu, H.Q., Lin, D.K.J., 2012. Uniform fractional factorial designs. Ann. Statist. 40, 891–907.
Wang, Y.P., Sun, F.S., Xu, H.Q., 2020. On design orthogonality, maximin distance and projection uniformity for computer experiments. J. Amer. Statist. Assoc. http://dx.doi.org/10.1080/01621459.2020.1782221.
Wang, J.C., Wu, C.F.J., 1992. Nearly orthogonal arrays with mixed levels and small runs. Technometrics 34, 409–422.
Wang, L., Xiao, Q., Xu, H.Q., 2018. Optimal maximin L_1-distance latin hypercube designs based on good lattice point designs. Ann. Statist. 46, 3741–3766.
Woods, D.C., Lewis, S.M., 2016. Design of experiments for screening. In: Ghanem, R., Higdon, D., Owhadi, H. (Eds.), Handbook of Uncertainty Quantification. Springer, New York, pp. 1143–1185.
Xiao, Q., Wang, L., Xu, H.Q., 2019. Application of kriging models for a drug combination experiment on lung cancer. Stat. Med. 38, 236–246.
Xiao, Q., Xu, H.Q., 2017. Construction of maximin distance latin squares and related latin hypercube designs. Biometrika 104, 455–464.
Xiao, Q., Xu, H.Q., 2018. Construction of maximin distance designs via level permutation and expansion. Statist. Sinica 28, 1395–1414.
Xu, H.Q., 2002. An algorithm for constructing orthogonal and nearly-orthogonal arrays with mixed levels and small runs. Technometrics 44, 356–368.
Xu, H.Q., 2003. Minimum moment aberration for nonregular designs and supersaturated designs. Statist. Sinica 13, 691–708.
Xu, H.Q., Wu, C.F.J., 2001. Generalized minimum aberration for asymmetrical fractional factorial designs. Ann. Statist. 29, 1066–1077.
Zhou, Y.D., Fang, K.T., Ning, J., 2013. Mixture discrepancy for quasi-random point sets. J. Complexity 29, 283–301.
Zhou, Y.D., Xu, H.Q., 2014. Space-filling fractional factorial designs. J. Amer. Statist. Assoc. 109, 1134–1144.