

## 演化排列优化拉丁超立方设计方法

李国盛, 张锡, 文谦, 杨家伟, 张为华, 武泽平\*  
(国防科技大学空天科学学院, 湖南长沙 410073)

**摘要:**拉丁超立方设计是最常用的计算机试验设计方法之一, 针对现有拉丁超立方设计方法采样一次性且难以兼顾设计的空间均匀性和计算效率的问题, 提出了一种演化排列拉丁超立方试验设计方法。通过对小样本设计的演化、排列信息继承和扩充等操作, 以较小的计算量实现了样本的扩充与优化。此外, 所提方法可以兼顾现有样本和新增采样点之间的关系, 实现样本的序列扩充, 这在实际近似建模过程中十分方便。通过多组数值试验, 验证了本文方法在空间均匀性和计算效率等方面的优越性。

**关键词:**计算机试验设计; 优化拉丁方设计; 排列信息继承; 空间均匀性

中图分类号: O212.6 文献标志码: A 开放科学(资源服务)标识码(OSID):

文章编号: 1001-2486(2024)03-150-08



听语音  
与作者  
聊科研

## Evolution permutation optimal Latin hypercube design method

LI Guosheng, ZHANG Xi, WEN Qian, YANG Jiawei, ZHANG Weihua, WU Zeping\*

(College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China)

**Abstract:** Latin hypercube design is one of the most commonly used computer experimental design methods, in response to the problem of one-time sampling and difficulty in balancing spatial uniformity and computational efficiency in existing Latin hypercube design methods, an experimental design method of permutation evolution Latin hypercube was proposed. By evolving small-sample designs, inheriting and expanding permutation information, the expansion and optimization of samples were achieved with a relatively small computational effort. In addition, this method gives consideration to the relationship between existing samples and new added samples and realizes the sequence expansion of samples, which is very convenient in the actual approximate modeling process. Through several groups of numerical experiments, the advantages of this method in space filling quality and calculation efficiency were verified.

**Keywords:** designs of computer experiments; optimal Latin hypercube design; permutation information inheritance; space-filling quality

工程实践中, 计算机数值模拟通常用于表征复杂的现实过程<sup>[1-3]</sup>。但随着设计变量和仿真精度的增加, 系统的单次仿真将变得非常耗时。为了利用有限的计算资源, 在最大限度上探索系统的全局特性, 研究者提出了多种试验设计 (design of experiments, DOE) 方法<sup>[4-5]</sup>。DOE 可以在设计空间内生成均匀分布的样本, 对设计空间的有效探索<sup>[6-8]</sup>。其中, 拉丁超立方设计 (Latin hypercube design, LHD) 以其优秀的空间均匀性能被广泛应用于试验设计领域。

为提高 LHD 的性能, 研究人员将优化准则与排列算法相结合, 提出了一系列适用于不同条件下的优化拉丁超立方设计 (optimal LHD, OLHD) 方法。试验设计的均匀性可以采用优化准则进行

表征, 常用的准则主要有最大熵准则、最大距离准则和  $\phi_p$  准则等。例如: Morris 等利用模拟退火算法构造拉丁超立方设计<sup>[9]</sup>; Jin 等使用增强随机进化 (enhanced stochastic evolutionary, ESE) 算法来优化设计的空间均匀性能<sup>[10]</sup>; Viana 等通过平移传播 (translational propagation algorithm, TPA) 算法直接构建 LHD<sup>[11]</sup>; Zhu 等提出连续局部枚举 (successive local enumeration, SLE) 算法<sup>[12]</sup>; 张甜甜等基于数独分组提出一种新型 LHD 构造方法<sup>[13]</sup>; 陈浩等改进了分片拉丁超立方设计<sup>[14]</sup>。现有方法在低维小样本条件下效果较好, 但随着样本维度和数量的增加, 其空间均匀性和计算效率将急剧下降。此外, 该方法通常是一次性生成所有样本的单阶采样方法, 难以应用于随机性较

收稿日期: 2022-04-26

基金项目: 国家自然科学基金资助项目 (52005502); 国防科技大学科研计划资助项目 (ZK19-11)

第一作者: 李国盛 (1998—), 男, 安徽宿州人, 博士研究生, E-mail: Liguosheng126@163.com

\*通信作者: 武泽平 (1990—), 男, 山西吕梁人, 副研究员, 博士, 硕士生导师, E-mail: zeping123@nudt.edu.cn

大的实际工程中。

针对上述问题,本文采用 OLHD 方法生成初始样本,提出了演化排列拉丁超立方设计(evolution permutation LHD, ELHD)方法。应用序列扩充思想,保证了新增样本和现有样本之间的分布均匀性,以较小的计算代价实现设计的更新与优化,大幅提升实际建模工作效率。

### 1 优化拉丁超立方设计

针对维度为  $d$ 、样本数量为  $N$  的试验设计问题,采用 LHD 方法获得分布均匀的样本。LHD 的结果可以表示为一个  $N \times d$  大小的设计矩阵,该矩阵的每一行数据表示一个样本,每一列对应所属变量的不同取值,并且可以被视作是整数  $1 \sim N$  的排列。LHD 的设计矩阵存在  $(N!)^d$  种排列方式,这使得随机生成的 LHD 通常难以保持良好的空间均匀性能。

$\phi_p$  准则被广泛应用于评价样本空间均匀性能,其核心思想为确保样本点间最小距离最大,引入  $\phi_p$  准则改进 LHD 可大幅提高设计空间均匀性。将设计中第  $i$  个和第  $j$  个采样点之间的距离表示为  $d_{ij}(1 \leq i < j \leq n)$ ,则  $\phi_p$  定义为:

$$\phi_p = \left( \sum_{i=1}^{s-1} \sum_{j=i+1}^s d_{ij}^{-p} \right)^{1/p} \quad (1)$$

式中,  $n$  为采样点个数,  $p$  为正整数(本文中  $p = 50$ ),  $s$  为不同距离值的数量。

考虑一个  $N \times d$  的试验设计问题,样本集  $A = \{S_1, S_2, \dots, S_N\}$  的矩阵形式如式(2)所示,  $S_i$  表示第  $i$  个样本点。

$$A = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1d} \\ x_{21} & x_{22} & \dots & x_{2d} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nd} \end{bmatrix} \quad (2)$$

采用  $\phi_p$  准则的 OLHD 方法的一般步骤可总结为:

- 步骤 1: 随机生成初始化设计  $L$ ;
- 步骤 2: 在  $L$  上执行不同的变换操作从而得到一系列新的设计;
- 步骤 3: 计算设计的  $\phi_p$  准则值,选择性能最优的结果作为当前最优解;
- 步骤 4: 重复前 3 个步骤,直到满足终止条件。

在处理低维小样本问题时,OLHD 方法具有良好的表现。但随着变量维数  $d$  和样本数量  $N$  的增加,OLHD 的计算复杂度  $(N!)^d$  将急剧增加,其效率和空间均匀性能也会大幅降低。针对

OLHD 在高维大样本条件下性能不理想的问题,提出基于演化排列方法更新和优化现有设计的 ELHD 方法,该方法在继承初始设计排列信息的同时可有效降低设计所需计算量。

### 2 演化排列改进方法

针对样本数量较大的试验设计,采用序列扩充方法可以有效提高采样效率。在样本扩充过程中,性能良好的初始样本包含的排列信息可为后续设计提供参考<sup>[15]</sup>。基于此,本节提出了一种继承初始样本排列信息,并在该信息指导下完成序列扩充的演化排列拉丁超立方设计方法。

#### 2.1 算法流程

在 ELHD 中,通过对初始设计执行演化操作以获得扩充所需的种子样本,并采用填充操作和规模调整步骤以满足设计的数量需求。此外,使用排列信息继承算法以较小的计算量更新和优化当前设计。ELHD 的流程图如图 1 所示,其具体步骤如下,并在第 2.2 节和第 2.3 节中给出相应步骤的详细说明。

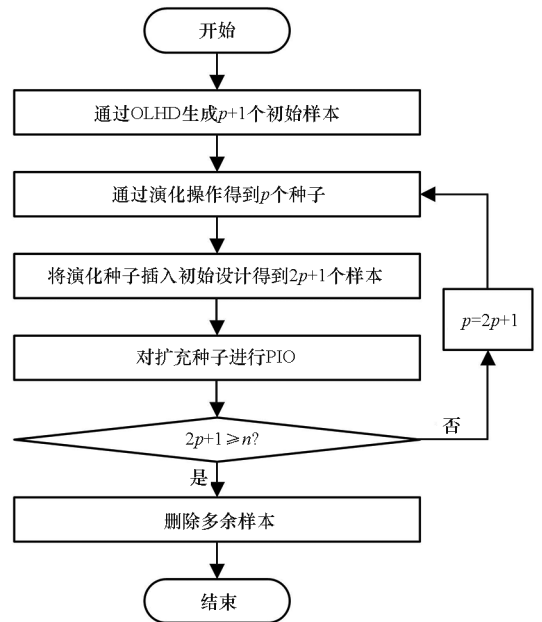


图 1 ELHD 方法流程图

Fig. 1 Flow chart of the ELHD method

步骤 1: 利用 OLHD 方法得到  $p + 1$  个均匀分布的样本作为初始样本集  $A_1 = \{S_1, S_2, \dots, S_p, S_{p+1}\}$ , 其中  $S_i$  表示第  $i$  个样本。

步骤 2: 通过对  $A_1$  进行演化操作得到演化矩阵  $A_{1best}^*$ 。

步骤 3:  $A_1$  中的  $p$  个间隔将形成一个  $p \times d$  的

试验空间  $B_1$ 。将演化矩阵  $A_{1best}^*$  叠加到试验空间  $B_1$  中,得到扩充矩阵  $A_1B_1$ 。

**步骤 4:**对  $A_1B_1$  进行 2.3 节中的基于排列信息继承的更新操作,保持  $A_{1best}^*$  各列间的变换关系不变,以  $A_{1best}^*$  的第一维排列为设计变量,以扩充矩阵  $A_1B_1$  的  $\phi_p$  值为目标函数进行优化,得到整体样本均匀性优良的设计。

**步骤 5:**判断样本数量  $2p + 1$  与需求的样本数量  $n$  的大小关系,若  $2p + 1 < n$  则令  $p = 2p + 1$ ,重复步骤 2 到步骤 5 直至  $2p + 1 \geq n$ 。

**步骤 6:**进行样本规模调整操作,除去冗余点,完成试验设计。

### 2.2 演化、扩充和数量调整算法

#### 2.2.1 初始设计

OLHD 的构造可采用多种算法,其中 ESE 算法的空间均匀性在中小型设计中具有明显优势<sup>[2,16]</sup>。选用 ESE 算法构建含有  $p + 1$  个样本的初始设计  $A_1$ ,样本矩阵为:

$$A_1 = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{(p+1)1} & x_{(p+1)2} & \cdots & x_{(p+1)d} \end{bmatrix} \quad (3)$$

$A_1$  中的  $p + 1$  个样本点将每个维度的设计域划分为  $p$  个区间,这些区间将形成一个新的设计试验空间  $B_1$ 。样本的空间一维投影如图 2 所示,需要  $p$  个种子实现对试验空间  $B_1$  的填充。

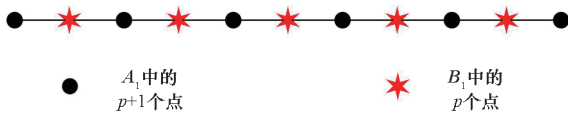


图 2 样本一维投影示意图

Fig.2 One dimensional projection diagram of samples

#### 2.2.2 演化操作

对初始设计执行演化操作以获得填充所需的  $p$  个种子。由于初始样本和种子的数量之差仅为 1,通过移除初始样本中的某个特定点来生成种子是较为便捷的方法。演化是指遍历初始样本点,使得去除该点后剩余样本的空间均匀性能最佳。通过该操作,实现初始样本排列信息的继承,从而降低设计所需计算量,具体步骤如下:

**步骤 1:**删除初始设计样本点,以获得演化矩阵  $A_{li}^*$ 。

$$A_{li}^* = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ \vdots & \vdots & & \vdots \\ x_{(i-1)1} & x_{(i-1)2} & \cdots & x_{(i-1)d} \\ x_{(i+1)1} & x_{(i+1)2} & \cdots & x_{(i+1)d} \\ \vdots & \vdots & & \vdots \\ x_{(p+1)1} & x_{(p+1)2} & \cdots & x_{(p+1)d} \end{bmatrix} \quad (4)$$

**步骤 2:**遍历所有样本点,以  $\min \phi_p$  为目标,从所有演化矩阵  $A_{li}^*$  中选出最优演化矩阵  $A_{1best}^*$ 。

$$A_{1best}^* = \begin{bmatrix} x'_{11} & x'_{12} & \cdots & x'_{1d} \\ x'_{21} & x'_{22} & \cdots & x'_{2d} \\ \vdots & \vdots & & \vdots \\ x'_{p1} & x'_{p2} & \cdots & x'_{pd} \end{bmatrix} \quad (5)$$

#### 2.2.3 叠加操作

将  $A_{1best}^*$  中的样本种子叠加到试验空间  $B_1$  中,得到扩展矩阵  $A_1B_1$ :

$$A_1B_1 = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x'_{11} & x'_{12} & \cdots & x'_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & & \vdots \\ x'_{p1} & x'_{p2} & \cdots & x'_{pd} \\ x_{(p+1)1} & x_{(p+1)2} & \cdots & x_{(p+1)d} \end{bmatrix} \quad (6)$$

#### 2.2.4 样本规模调整

若在最后一个循环中所获样本数量大于所需数量,则根据文献[11]中的方法删除距离当前设计空间中心最远的冗余点。

### 2.3 排列信息继承算法

多种 OLHD 方法采用列元素交换更新设计。列元素交换是指在设计矩阵的同一列中交换两个不同的元素<sup>[11,16]</sup>。然而,随着样本规模和维度的增加,列元素交换效率制约着试验设计速度。针对该问题,提出一种基于排列信息继承操作(permutation information inheritance operation, PIO)的列操作方法用于更新和优化设计矩阵。

LHD 矩阵的任意两列  $P_i$  和  $P_j$  均是满足  $X^l \leq x \leq X^u$  的整数排列,  $X^l$  和  $X^u$  分别为设计区间的上下限。

$$\begin{cases} P_i = [x_{1i}, x_{2i}, \dots, x_{Ni}]^T \\ P_j = [x_{1j}, x_{2j}, \dots, x_{Nj}]^T \end{cases} \quad (7)$$

结合线性代数的相关知识可知,任意  $P_j$  均可由  $P_i$  经式(8)变换得到。

$$P_j = T_{ij}P_i \quad (8)$$

式中,  $T_{ij}$  是大小为  $N \times N$  的变换矩阵。

通过线性变换,LHD 的各列都可与其第  $i$  列相关联。若设置  $i = 1$ ,则整个设计与矩阵第一列相关联。在保持变换矩阵  $T_{ij}$  不变的情况下,通过对 LHD 所得矩阵的第一列执行列元素交换,可以实现整个设计的更新,该操作即为排列信息继承操作。以大小为  $5 \times 4$  的设计为例,对其进行排列信息继承操作的结果如图 3 所示。

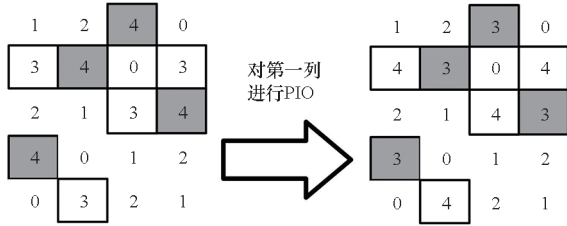


图3 PIO示意图  
Fig.3 PIO diagram

显然,在 PIO 之后,LHD 的每一列都发生了变化,而设计仍为 LHD。因此,在试验空间  $B_1$  中执行 PIO,并保持扩展矩阵  $A_1B_1$  中初始样本的位置不变。通过继承初始设计的排列属性,以较低的计算成本实现整个设计的更新。在所有结果中,选择空间均匀质量最好的一个作为优化后的样本  $A_1A_2$ :

$$A_1A_2 = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{11}^o & x_{12}^o & \cdots & x_{1d}^o \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & & \vdots \\ x_{p1}^o & x_{p1}^o & \cdots & x_{p1}^o \\ x_{(p+1)1} & x_{(p+1)2} & \cdots & x_{(p+1)d} \end{bmatrix} \quad (9)$$

式中,  $x_{ii}^o$  表示空间  $B_1$  中种子的优化排列。

通过该操作,遍历更新 LHD 的计算量由原先的  $(N!)^d$  下降至  $(N!)$ ,后续章节将对 PIO 的效果进行验证。二维 ELHD 构建过程如下:

**步骤 1:**由 ESE 算法生成样本数量为 11 的初始设计  $A_1$ ,如图 4(a)所示。

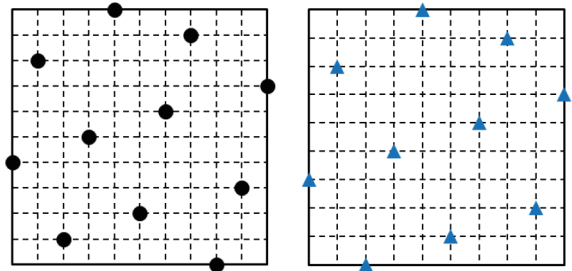
**步骤 2:**对初始设计执行演化操作,得到样本数量为 10 的最优演化设计  $A_{1best}^*$ ,如图 4(b)所示。

**步骤 3:**通过叠加操作得到扩展矩阵  $A_1B_1$ ,如图 4(c)所示,其中蓝色三角表示叠加的种子,黑色圆点表示已有的采样点。

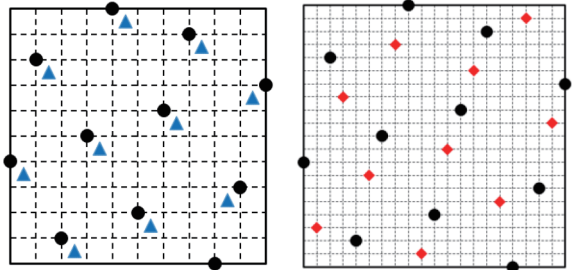
**步骤 4:**执行 PIO 以构建优化后的设计  $A_1A_2$ ,如图 4(d)所示。重复上述过程以获得第二和第三个循环的结果,如图 4(e)和图 4(f)所示,其中红点表示在该循环中获得的新采样点,黑点表示

现有采样点。

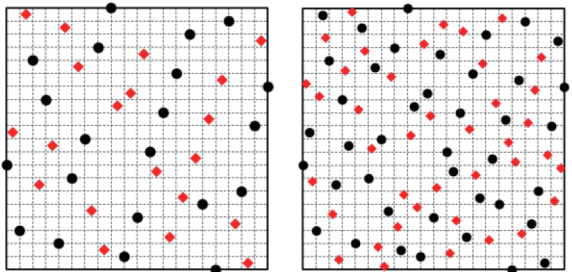
**步骤 5:**删除冗余点以获得所需的设计。



(a) 初始设计  
(a) Initial design  
(b) 演化设计  
(b) Evolutionary design



(c) 扩充矩阵  
(c) Extended matrix  
(d) 第一轮扩充  
(d) The first expansion



(e) 第二轮扩充  
(e) The second expansion  
(f) 第三轮扩充  
(f) The third expansion

图4 ELHD 的二维示例  
Fig.4 2D example of ELHD

与采用直接优化方法相比,ELHD 在扩充样本时实现了新样本与原有样本整体的均匀性。通过排列信息继承算法,将整个 LHD 的排列与其第一列关联,极大减少了更新样本的计算量。

合理选择初始样本的大小,给定参数  $p$  的取值范围,仍需要做进一步讨论。当  $p$  过小时,初始样本所包含的排列信息不够全面,容易出现明显的偏差趋势;当  $p$  过大时,会给设计的优化求解带来困难。所以,下面将结合数值试验对参数  $p$  的影响进行讨论。

### 2.4 参数 $p$ 的影响

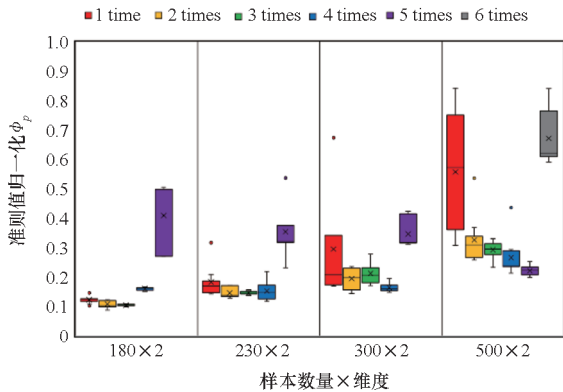
为了分析参数  $p$  的影响,进行维度从 2 到 10、样本数量从 180 到 500 的一系列试验。

为避免 2.2 节中样本规模的调整过程对结果产生影响,在给定设计需求后,通过倒推的方式选择初始样本的大小。例如,要完成一个大小为 100 的设计,则可选取的初始样本大小为:51、26 和 14 等,与之对应的扩充次数分别为:1 次、2 次和 3 次。重复进行 20 次试验以降低随机因素所造成的影响。此外,为了便于比较,利用式 (10) 对  $\phi_p$  值进行归一化。

$$\overline{\phi_p} = \frac{\phi_p - \min \phi_p}{\max \phi_p - \min \phi_p} \quad (10)$$

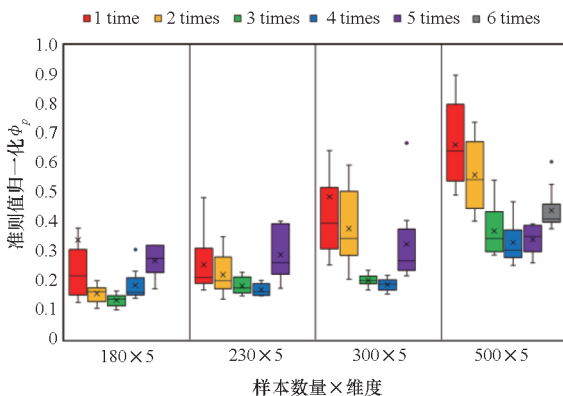
其中,  $\max \phi_p$  和  $\min \phi_p$  分别表示同组试验所获取  $\phi_p$  的最大值和最小值。

所得结果如图 5 所示,箱型图的值越小越集中,对应结果的性能越均匀越稳健。在每个系列中均有一个明显的转折点,同一系列中,扩充次数的增加对应着参数  $p$  数值的减小。可以看出,随着  $p$  的减小,ELHD 的稳健性和空间均匀性能呈现出先优后劣的趋势,与 2.3 节中的分析一致。表 1 总结了图 5 中每个系列表现最优的  $p$  值。从该表可以看出,当  $p \in [14, 28]$  时,可获得最佳结果。



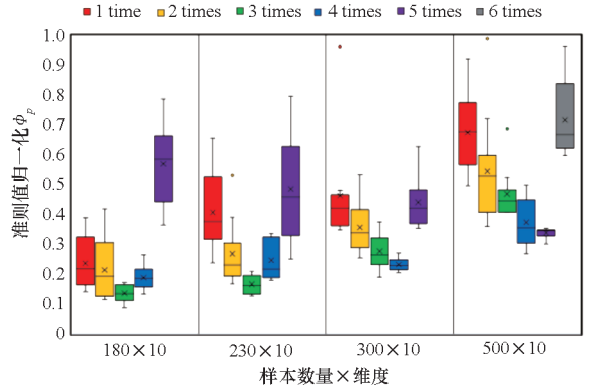
(a) 2 维样本的箱式图

(a) Box diagram of 2D sample



(b) 5 维样本的箱式图

(b) Box diagram of 5D sample



(c) 10 维样本的箱式图

(c) Box diagram of 10-d sample

图 5  $\phi_p$  归一化后的箱型图

Fig. 5 Box diagram of the normalized  $\phi_p$

表 1 同一系列中性能最佳的参数  $p$

Tab. 1 The best performance  $p$  in the same series

样本维度	样本数量	扩充次数	$p$
2	180	3	22
	230	3	28
	300	4	18
	500	5	15
5	180	3	22
	230	4	14
	300	4	18
	500	5	15
10	180	3	22
	230	3	28
	300	4	18
	500	5	15

### 3 数值试验

本节通过数值算例,比较 ESE、TPA 和 lhsdesign 三种算法与 ELHD 算法在样本空间均匀质量和计算效率上的差异。以上所有算法均是在 Windows 10 下使用配备 Intel Core i7-9750H 2.6 GHz CPU、16 GB RAM 和 MATLAB R2019b 的个人计算机上运行。为降低随机影响,以每种算法独立运行 50 次所获得的平均值作为结果。

#### 3.1 空间均匀性能比较

为探索 ELHD 的空间均匀性能,进行了一系列试验。LHD 的采样设置如表 2 所示,本节考虑了 Viana 等<sup>[11]</sup>提出的采样设置问题。ELHD 以小样本 OLHD 所得结果为初始设计,结合序列扩充思想后该方法更适用于大中型样本设计。由于高

维条件下(维度达到 10 维及以上)ESE 的计算时间过长,本节主要研究维度范围为 2 到 8 的大中型采样问题。

表 2 LHD 的采样设置

Tab.2 Sampling settings of LHD

样本 维度	样本数量		
	小型采样问题	中型采样问题	大型采样问题
2	12	20	120
4	30	70	300
6	56	168	560
8	90	330	900
10	132	572	1 320
12	182	910	1 820

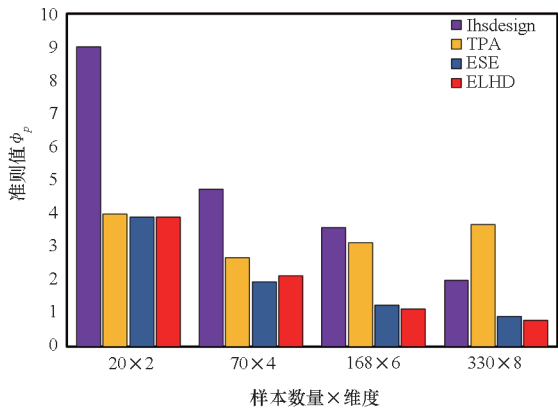
图 6 给出了大中型采样问题空间均匀性性能的比较结果。可以看出,lhsdesign 的性能最差且随机性较强。TPA 在维度较低且样本数量较大时表现良好,随着维度的增加,其竞争力下降。在大多

数情况下,ESE 与 ELHD 有着相似的空间均匀性,且明显优于其他两种方法。ELHD 在绝大多数问题上有着较优的表现,其优势随着样本数量和维度的增加而更加明显。

### 3.2 计算效率比较

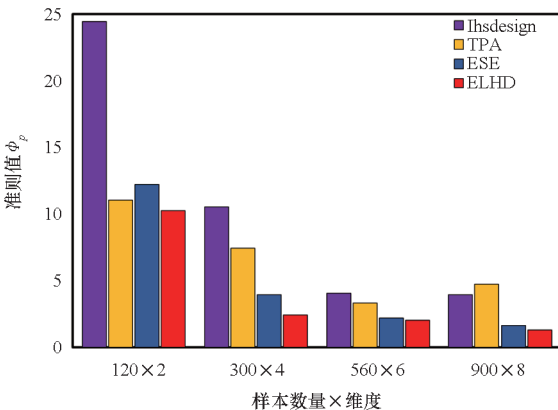
由于 ELHD 和 ESE 的空间均匀质量在大多数情况下较为接近,且 ELHD 方法的初始样本通过 ESE 算法生成,对 ELHD 和 ESE 的计算效率进行对比,以验证 ELHD 性能。

图 7 显示了不同维度下( $d=2,5,10,20$ ),样本数量为 100 和 200 时计算效率的比较结果。从图 7 可以看出,得益于 LHD 的直接构造,TPA 和 lhsdesign 相较于 ELHD 耗时更少。然而,TPA 和 lhsdesign 的空间均匀性往往并不令人满意。在满足良好均匀性的前提下,ELHD 的计算效率要明显优于 ESE。并且,随着维数的增加,ELHD 的优势将愈发明显。



(a) 中型采样问题的比较

(a) Comparison of medium-scale sampling problem

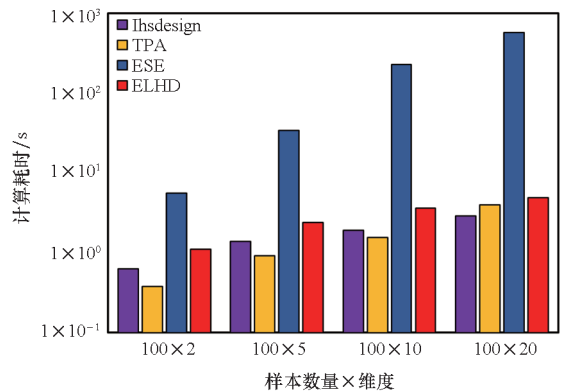


(b) 大型采样问题的比较

(b) Comparison of large-scale sampling problem

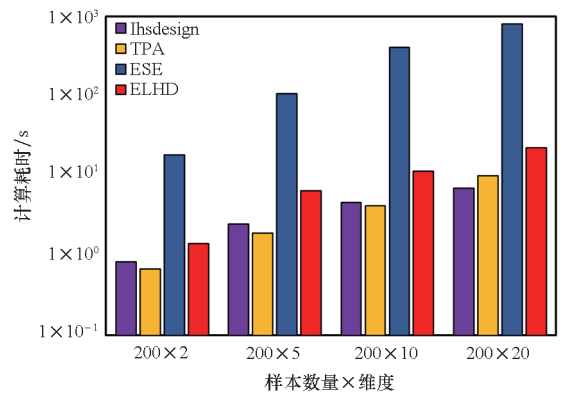
图 6 大中型采样问题对比

Fig.6 Comparison of large and medium sampling problems



(a) 不同维度下 100 个样本的比较

(a) Comparison of 100 samples with different dimensions



(b) 不同维度下 200 个样本的比较

(b) Comparison of 200 samples with different dimensions

图 7 不同维度下算法耗时的比较

Fig.7 Comparison of algorithm time consumption in different dimensions

为了进一步体现 ELHD 的高效性,下面进行

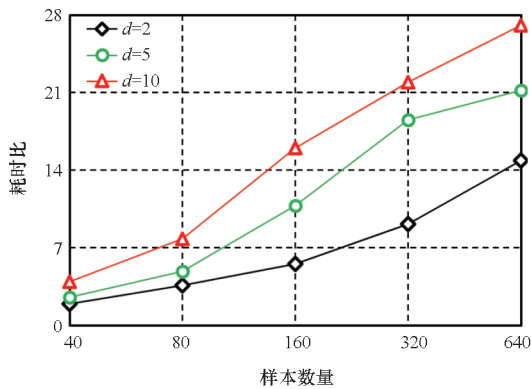
给定初始样本条件下的扩充试验。计算效率体现为同等条件下完成设计所需的工作时间,并通过式(11)计算上述两种方法的耗时比。

$$R = \frac{T_{ESE}}{T_{ELHD}} \quad (11)$$

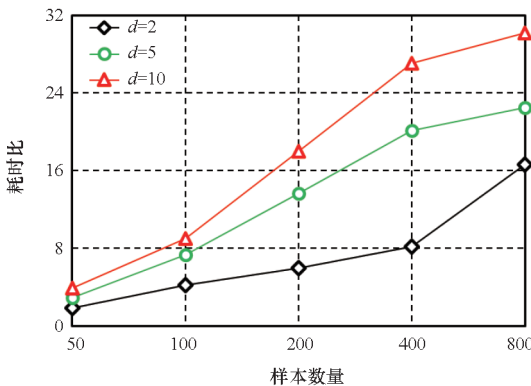
式中,  $T_{ESE}$  和  $T_{ELHD}$  分别表示两种算法的计算时间。

设置参数  $p$  分别为 20 和 25, 在不同维度 ( $d=2, 5, 10$ ) 上进行序列扩充, 同等条件下所获得的耗比如图 8 所示。

结果表明, 相较于 ESE, ELHD 的计算效果与样本数量和维度正相关。随着样本数量的增加, ELHD 的效率显著提高。此外, 在样本数量相同的情况下, 维数越高, ELHD 的优势越明显。尤其针对  $800 \times 10$  的设计, ELHD 的计算效率较 ESE 提升约 96%。ELHD 兼具空间均匀性能及计算效率, 在大中型采样问题中具有显著优势。这主要得益于 ELHD 通过演化、扩充等操作可以直接实现样本数量的序列扩增。并且, 基于排列信息继承操作, 整个设计的更新优化问题转变为初始排列信息指导下的单列优化问题, 使得优化整个设计的计算复杂度显著降低, 从而进一步提高了设计效率。



(a)  $p = 20$



(b)  $p = 25$

图 8 耗时对比图

Fig. 8 Comparison of time-consuming ratio

## 4 结论

本文提出了一种基于演化排列算法的拉丁超立方设计方法。该方法采用排列信息继承算法, 以较小的计算代价实现了对整个设计的更新和优化。采用数值算例对 ELHD 与其他 3 种常见方法的空间均匀质量进行对比, 再选择空间均匀性较好的两种方法进行计算效率的研究。在 8 个大中型采样问题中进行了空间均匀性能的比较, 结果显示, ELHD 的空间均匀性能明显优于 lhdesign、TPA 两种算法, 在大部分算例中与 ESE 性能相近。其后, 针对同一条件下 ELHD 与 ESE 的计算效率进行对比。结果表明, ELHD 的效率明显优于 ESE, 其优势随着样本维度和数量的增加而提升, 在设计为  $800 \times 10$  时, ELHD 耗时仅需 ESE 的 4% 左右。综上所述, ELHD 是进行大中型采样的有效方法。并且, 其所具有的动态扩充特性使其在实际工程应用中可以更灵活地选择样本数量, 从而避免了过拟合以及计算资源浪费等问题, 相较于其他一次性采样方法具有明显优势。

## 参考文献 (References)

- [1] 刘晓路, 陈英武, 荆显荣, 等. 优化拉丁方试验设计方法及其应用[J]. 国防科技大学学报, 2011, 33(5): 73-77.  
LIU X L, CHEN Y W, JING X R, et al. Optimized Latin hypercube sampling method and its application[J]. Journal of National University of Defense Technology, 2011, 33(5): 73-77. (in Chinese)
- [2] 刘新亮, 郭波. 适用于复杂系统仿真试验的试验设计方法[J]. 国防科技大学学报, 2009, 31(6): 95-99.  
LIU X L, GUO B. Studies on the optimization of Latin hypercube designs for complex system simulation[J]. Journal of National University of Defense Technology, 2009, 31(6): 95-99. (in Chinese)
- [3] 徐璿, 段晓君, 王正明, 等. 灵活的多层嵌套拉丁超立方体设计构造[J]. 国防科技大学学报, 2019, 41(3): 174-178.  
XU J, DUAN X J, WANG Z M, et al. A new construction for flexible multi-layer nested Latin hypercube designs [J]. Journal of National University of Defense Technology, 2019, 41(3): 174-178. (in Chinese)
- [4] SIMPSON T W, BOOKER A J, GHOSH D, et al. Approximation methods in multidisciplinary analysis and optimization: a panel discussion [J]. Structural and Multidisciplinary Optimization, 2004, 27(5): 302-313.
- [5] SIMPSON T W, POPLINSKI J D, KOCH P N, et al. Metamodels for computer-based engineering design: survey and recommendations [J]. Engineering with Computers,

- 2001, 17(2): 129 - 150.
- [6] 杜丽, 吕利叶, 孙伟, 等. 一种适用于约束空间的拉丁超立方取点策略[J]. 机械设计与制造, 2021(8): 43 - 47.  
DU L, LYU L Y, SUN W, et al. An Latin hypercube sampling approach for constrained design space [J]. Machinery Design & Manufacture, 2021(8): 43 - 47. (in Chinese)
- [7] 文谦, 杨家伟, 武泽平, 等. 固体姿轨控发动机喉栓运动时序实验设计方法[J]. 推进技术, 2022, 43(9): 354 - 363.  
WEN Q, YANG J W, WU Z P, et al. Time-sequence experiment design for pintle motion scheme of solid divert and attitude control system[J]. Journal of Propulsion Technology, 2022, 43(9): 354 - 363. (in Chinese)
- [8] SUN Y B, MENG X Y, LONG T, et al. A fast optimal Latin hypercube design method using an improved translational propagation algorithm[J]. Engineering Optimization, 2020, 52(7): 1244 - 1260.
- [9] MORRIS M D, MITCHELL T J. Exploratory designs for computational experiments[J]. Journal of Statistical Planning and Inference, 1995, 43(3): 381 - 402.
- [10] JIN R C, CHEN W, SUDJANTO A. An efficient algorithm for constructing optimal design of computer experiments [J]. Journal of Statistical Planning and Inference, 2016, 134(1): 268 - 287.
- [11] VIANA F A C, VENTER G, BALABANOV V. An algorithm for fast optimal Latin hypercube design of experiments[J]. International Journal for Numerical Methods in Engineering, 2010, 82(2): 135 - 156.
- [12] ZHU H G, LIU L, LONG T, et al. A novel algorithm of maximin Latin hypercube design using successive local enumeration[J]. Engineering Optimization, 2012, 44(5): 551 - 564.
- [13] 张甜甜, 李妮, 龚光红, 等. 一种基于数独分组的拉丁超立方试验设计方法[J]. 系统仿真学报, 2020, 32(11): 2185 - 2191.  
ZHANG T T, LI N, GONG G H, et al. A fast Latin hypercube experiment design method based on sudoku grouping[J]. Journal of System Simulation, 2020, 32(11): 2185 - 2191. (in Chinese)
- [14] 陈浩, 张艳. 投影均匀分片拉丁超立方体设计[J]. 系统科学与数学, 2020, 40(2): 366 - 374.  
CHEN H, ZHANG Y. Uniform projection sliced Latin hypercube designs [J]. Journal of Systems Science and Mathematical Sciences, 2020, 40(2): 366 - 374. (in Chinese)
- [15] WU Z P, WANG D H, OKOLO P N, et al. Efficient space-filling and near-orthogonality sequential Latin hypercube for computer experiments [J]. Computer Methods in Applied Mechanics and Engineering, 2017, 324: 348 - 365.
- [16] VIANA F A C. A tutorial on Latin hypercube design of experiments [J]. Quality and Reliability Engineering International, 2016, 32(5): 1975 - 1985.