Construction of Nested Orthogonal Arrays

1 Introduction

In many engineering and scientific disciplines, it could be costly, time-consuming or even infeasible to investigate a complex physical process by conventional experimental approaches. As computing power has increased, deterministic computer models have already become prevalent surrogates for efficiently studying these sophisticated real-world systems. In practice, Gaussian random function models are used to build surrogate models by interpolating all the observed responses, and system outputs over the entire input domain can then be readily emulated. This enables an investigator to systematically explore the relationship between the input and output variables of a complex system. For an excellent introduction to the methodology of designing, modeling and analyzing computer experiments, the reader is referred to Santner et al. (2003) and Fang et al. (2006). Typically, design and analysis strategies are all developed for computer simulations with a single level of accuracy. In practice, however, a large-scale computer code might require several hours or even a few days to return a single response. To build surrogate models under limited computation resources, complex computer codes are frequently implemented with two levels of accuracy, such that the high-accuracy simulations are more accurate but computationally expensive, and the low-accuracy simulations are less accurate but inexpensive in computation. The input-output relationships can then be explored through a surrogate model built by adequately integrating the system responses derived from both high- and low-accuracy simulations.

To efficiently execute deterministic computer codes with various levels of accuracy, an innovative planning scheme is therefore required and has received much attention in the past few years. To the best of my knowledge, Qian et al. (2009b) first proposed a new class of nested space-filling designs called the nested orthogonal array-based Latin hypercube designs, abbreviated as NOA-based LHDs hereinafter, for planning multi-fidelity computer simulations. For a given nested orthogonal array, a NOA-based LHD can be readily derived through the construction method developed by Tang (1993) with the modified labeling scheme introduced by Qian et al. (2009b). When a nested space-filling design is carried out, the differences between the system outputs with various fidelity levels can be observed. The responses obtained from the high-accuracy simulations can then be utilized to calibrate the surrogate model built by the low-accuracy simulations, such that the emulator of system response can achieve satisfactory prediction accuracy. For different construction methods of nested orthogonal arrays, the reader is referred to Qian et al. (2009a), Qian et al. (2009b), Dey (2010), Sun et al. (2013), and Sun et al. (2014), among others. Under the constraints of limited computation resources, a series of candidate designs with run-size flexibility is usually preferred by a practitioner. However, none of the aforementioned studies offers a flexible choice in run-size. In this article, a new class of nested orthogonal arrays called the nested orthogonal arrays of parallel-flats type is therefore introduced for addressing this practical issue. A noteworthy feature of the proposed designs is their run-size flexibility. For a given number of input variables of interest, the proposed methods can generate an extensive collection of candidate designs with different run-sizes. This allows a researcher to choose a planning scheme for a series of two-fidelity computer simulations subject to a constraint on run-size.

2 Notation and Definitions

Firstly, some notation, definitions and terms to be used throughout this article are introduced in this section.

2.1 Galois Field Projections

Let p be a prime, then the set of residues $\{0, 1, \ldots, p-1\}$ modulo p comprises a Galois field denoted by $GF(p)$ under the addition and multiplication modulo p. Furthermore, let $f(x) = a_u x^u + a_{u-1}x^{u-1} + \cdots + a_1 x + a_0$ be an irreducible polynomial of degree u, where $a_i \in GF(p)$ for all i; and $a_u = 1$. Under the addition and multiplication of polynomial modulo $f(x)$, the set of all polynomials of degrees less than u comprises a Galois field denoted by $GF(p^u)$. Throughout, let $s_1 = p^{u_1}$ and $s_2 = p^{u_2}$ be two distinct powers of the same prime p, where the integers $u_1 > u_2 \geq 1$. Let F_1 denote the $GF(s_1)$ with irreducible polynomial $f_1(x)$, and F_2 denote the $GF(s_2)$ with irreducible polynomial $f_2(x)$. Accordingly, a Galois field projection introduced by Bose and Bush (1952), which associates an element on F_1 with another element on F_2, is presented as follows. For any $q(x) = a_{u_1-1}x^{u_1-1} + \cdots + a_{u_2-1}x^{u_2-1} + \cdots + a_1 x + a_0 \in F_1$, the projection $\phi\left[q(x)\right]$ is defined as

$$\phi\left[q(x)\right] = a_{u_2-1}x^{u_2-1} + \cdots + a_1 x + a_0.$$

Clearly, the projection ϕ eliminates all the terms of degree u_2 and higher, such that the output is an element of F_2. There are some other projections defined on Galois fields which play a prominent role in the construction of nested orthogonal arrays. For further discussions regarding different projections on Galois fields and their applications, the reader can consult Qian et al. (2009a) and Sun et al. (2014). Below, two examples are given to illustrate the Galois field projections used in this article.

Example 1. Let $p = 2$, $u_1 = 2$ and $u_2 = 1$, then $s_1 = 4$ and $s_2 = 2$. Accordingly, $F_1 = \{0, 1, x, x+1\}$ is a $GF(4)$ with irreducible polynomial $x^2 + x + 1$, and $F_2 = \{0, 1\}$ is a $GF(2)$ with irreducible polynomial $x + 1$. For associating the elements on these two Galois fields, the projection $\phi\left[q(x)\right]$ is defined as follows:

$$\phi[q(x)] = \begin{cases} 0 & \text{if } \; q(x) = 0 \; \text{ or } \; x. \\ 1 & \text{if } \; q(x) = 1 \; \text{ or } \; x+1. \end{cases}$$

Clearly, $\phi\left[q(x)\right]$ associates an element on $GF(4)$ with an alternative one on $GF(2)$. □

Example 2. Let $p = 2$, $u_1 = 3$ and $u_2 = 1$, then $s_1 = 8$ and $s_2 = 2$. It is straightforward to know that $F_1 = \{0, 1, x, x+1, x^2, x^2+1, x^2+x, x^2+x+1\}$ is a $GF(8)$ with irreducible polynomial x^3+x+1, and $F_2 = \{0, 1\}$ is a $GF(2)$ with irreducible polynomial $x+1$. The Galois field projection $\phi\,[q(x)]$ is then defined as follows:

$$\phi[q(x)] = \begin{cases} 0 & \text{if } q(x) = 0 \text{ or } x \text{ or } x^2 \text{ or } x^2+x. \\ 1 & \text{if } q(x) = 1 \text{ or } x+1 \text{ or } x^2+1 \text{ or } x^2+x+1. \end{cases}$$

Obviously, $\phi\,[q(x)]$ maps a polynomial on $GF(8)$ to another one on $GF(2)$. □

2.2 Parallel-Flats Designs and Orthogonal Arrays

For a prime p and a positive integer u, let $GF(s)$ denote the Galois field of order s, where $s = p^u$, the uth power of the prime p. Furthermore, let $\boldsymbol{T}_i$ be an $s^{n-k} \times n$ matrix, whose rows represent the s^{n-k} solutions $\boldsymbol{t}$ of linear equations $\boldsymbol{A}\boldsymbol{t} = \boldsymbol{c}_i$ over $GF(s)$ for $i = 1, \ldots, f$. Specifically, $\boldsymbol{A}$ is a $k \times n$ matrix of rank k, and $\boldsymbol{c}_i$ is a $k \times 1$ vector. Typically, $\boldsymbol{T}_i$ is called a single-flat design. By juxtaposing f single-flat designs $\boldsymbol{T}_1, \ldots, \boldsymbol{T}_f$, an $N \times n$ matrix $\boldsymbol{L}$ can be obtained easily, where $N = f \times s^{n-k}$. The matrix $\boldsymbol{L}$ is commonly called a parallel-flats design determined by the matrix pair $(\boldsymbol{A}, \boldsymbol{C})$, where $\boldsymbol{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_f]$, a $k \times f$ matrix. Clearly, parallel-flats designs can be constructed by collecting all the solutions of linear equations determined by the matrix pair $(\boldsymbol{A}, \boldsymbol{C})$ over Galois fields. The construction of parallel-flats designs dates back to the pioneering work of Connor and Young (1959) and later studied by several researchers, including Addelman (1961) and John (1962), among others. For an overview of the general theory of parallel-flats designs, the reader is referred to Cheng (2014).

An $N \times n$ matrix $\boldsymbol{L}$ with entries from a set of s distinct elements is said to be an s-symbol orthogonal array of strength t denoted by $OA(N, n, s, t)$, if all possible combinations of the s elements occur equally often as rows in any $N \times t$ submatrix of $\boldsymbol{L}$. As an immensely important tool for planning multifactorial experiments, orthogonal arrays have been successfully employed in industrial process improvements, quality control, biopharmaceutical studies, clinical trails, and many other scientific disciplines. For a comprehensive introduction to the theory, construction and applications of orthogonal arrays, the reader can consult Hedayat et al. (1999) and Wu and Hamada (2009). Under the framework of parallel-flats designs, Srivastava and Throop (1990) proposed the sufficient and necessary condition for characterizing the required matrix pair $(\boldsymbol{A}, \boldsymbol{C})$, such that the resulting $\boldsymbol{L}$ is an orthogonal array. Orthogonal arrays derived through this construction method are usually called the orthogonal arrays of parallel-flats type. This special class of orthogonal arrays is commonly used for tackling real-world problems, primarily due to its simple construction and run-size flexibility.

Let $\boldsymbol{U} = (u_{ij})$ and $\boldsymbol{V} = (v_{kl})$ be $r_1 \times c_1$ and $r_2 \times c_2$ matrices with entries from $GF(s)$. Subsequently, two operations for $\boldsymbol{U}$ and $\boldsymbol{V}$ are introduced for constructing the proposed designs. The first operation $\boldsymbol{U} \oplus \boldsymbol{V}$ is defined by

$$\boldsymbol{U} \oplus \boldsymbol{V} = \begin{bmatrix} u_{11} * \boldsymbol{V} & u_{12} * \boldsymbol{V} & \cdots & u_{1c_1} * \boldsymbol{V} \\ u_{21} * \boldsymbol{V} & u_{22} * \boldsymbol{V} & \cdots & u_{2c_1} * \boldsymbol{V} \\ \vdots & \vdots & \ddots & \vdots \\ u_{r_1 1} * \boldsymbol{V} & u_{r_1 2} * \boldsymbol{V} & \cdots & u_{r_1 c_1} * \boldsymbol{V} \end{bmatrix},$$

where $*$ stands for the addition on $GF(s)$; and $u_{ij} * \boldsymbol{V}$ denotes a matrix obtained by adding u_{ij} to all the elements of $\boldsymbol{V}$, that is, $u_{ij} * \boldsymbol{V} = (u_{ij} * v_{kl})$. Clearly, the resulting matrix of $\boldsymbol{U} \oplus \boldsymbol{V}$ is an $(r_1 r_2) \times (c_1 c_2)$ matrix. Similarly, the second operation $\boldsymbol{U} \otimes \boldsymbol{V}$ can be defined by replacing the addition with multiplication on $GF(s)$. Alternatively, let $\mathcal{G}$ be the additive group associated with the Galois field $GF(s)$. An $r \times c$ matrix $\boldsymbol{D}$ with entries from $\mathcal{G}$ is called a difference matrix of s symbols, if every element of $\mathcal{G}$ appears equally often in the vector difference between any two columns of $\mathcal{D}$. The concept of difference matrices was originally introduced by Bose and Bush (1952), and it was then widely utilized in the construction of orthogonal arrays. The reader is referred to Hedayat et al. (1999) for an excellent reference for difference matrices. Based on the construction method proposed by Bose and Bush (1952), if $\boldsymbol{D}$ is an $r \times c$ difference matrix of s symbols and $\boldsymbol{L}$ is an $OA(N, n, s, 2)$, then the following array

$$\boldsymbol{L} \oplus \boldsymbol{D}$$

is an $OA(Nr, nc, s, 2)$. This provides a simple but powerful method for generating large orthogonal arrays through existing small ones, and a similar idea is adopted by the construction methods presented in the subsequent sections.

3 Design Construction

Suppose that the projection ϕ is used in the level-collapsing scheme for $\boldsymbol{L}$, and the resulting matrix is denoted by $\phi(\boldsymbol{L})$. Firstly, the formal definition of a nested orthogonal array is explicitly given below.

Definition 1. The triplet $(\boldsymbol{L}_1, \boldsymbol{L}_2, \phi)$ constitutes a nested orthogonal array of strength t denoted by $NOA((N_1, N_2), n, (s_1, s_2), t)$ where $\boldsymbol{L}_2$ is a submatrix of $\boldsymbol{L}_1$ if and only if (a) the $N_1 \times n$ matrix $\boldsymbol{L}_1$ is an $OA(N_1, n, s_1, t)$; and (b) the $N_2 \times n$ matrix $\phi(\boldsymbol{L}_2)$ is an $OA(N_2, n, s_2, t)$.

For planning a series of computer simulations with two levels of accuracy, candidate designs with a flexible choice in run-size is usually preferred by a practitioner. For a given number of input variables n, some construction methods are introduced for generating nested orthogonal arrays with various run-sizes.

3.1 An Algorithm

Let $\boldsymbol{A}_1$ be an $s_1 \times (s_1 + 1)$ matrix given by

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{F}_1 & \boldsymbol{I}_{s_1} \end{bmatrix}. \tag{1}$$

When $\boldsymbol{A}_1$ is given as (1), parallel-flats designs with flexible run-sizes can be constructed easily, because of the fact that a single-flat design determined by this $\boldsymbol{A}_1$ has the smallest run-size s_1. When the $\boldsymbol{A}_1$ in (1) is chosen, a nested orthogonal array can be generated by searching an $s_1 \times f$ matrix $\boldsymbol{C}_1$, such that the conditions described in Corollary 1 are fulfilled. Before presenting the proposed algorithm, an interesting fact, which plays a prominent role in the computer search, is first introduced.

Lemma 1. For a given $\boldsymbol{A}_1$ as (1), the matrix pair $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ satisfies the condition (a) of Corollary 1 with $t \geq 2$ only if each row of $\boldsymbol{C}_1$ is uniform on $GF(s_1)$.

For generating a nested orthogonal array, according to Lemma 1, when $\boldsymbol{A}_1$ is given as (1), the rows of a candidate $\boldsymbol{C}_1$ should be requested to consist of the s_1 elements of $GF(s_1)$ equally often. This fact not only decreases the number of candidates, but also significantly decreases the computational cost in searching the required $\boldsymbol{C}_1$. Let $\mathcal{R}$ be the candidate set consisting of all uniform $1 \times f$ vector on $GF(s_1)$. Note that the number of flats f here should be equal to a multiple of s_1, that is, $f = g \times s_1$, where g is a positive integer, due to the requirement of uniformity for all vectors in $\mathcal{R}$. For generating the proposed designs, an algorithm is then devised below.

Step 0: Generate the candidate set $\mathcal{R}$.

Step 1: When g is odd, choose s_1 distinct row vectors $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_{s_1}$ from $\mathcal{R}$, and a candidate $\boldsymbol{C}_1$ is constructed by

$$\boldsymbol{C}_1 = \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \\ \vdots \\ \boldsymbol{r}_{s_1} \end{bmatrix}.$$

On the other hand, when g is even, choose z distinct row vectors $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_z$ from $\mathcal{R}$, and a candidate $\boldsymbol{C}_1$ is constructed by

$$\boldsymbol{C}_1 = \begin{bmatrix} \mathbf{1}_{s_2} \otimes \boldsymbol{r}_1 \\ \mathbf{1}_{s_2} \otimes \boldsymbol{r}_2 \\ \vdots \\ \mathbf{1}_{s_2} \otimes \boldsymbol{r}_z \end{bmatrix},$$

where $z = s_1/s_2 = p^{u_1-u_2}$, the ratio between s_1 and s_2.

Step 2: Verify whether the matrix pairs $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ satisfy the conditions described in Corollary 1, where $\boldsymbol{A}_1$ is given as (1); $\boldsymbol{A}_2 = \phi(\boldsymbol{A}_1)$; and $\boldsymbol{C}_2 = \phi(\boldsymbol{C}_1)$. If so, terminate the procedure. Otherwise, return to Step 1 for generating an alternative candidate $\boldsymbol{C}_1$.

When implementing the proposed algorithm, the run-sizes of obtained designs for the two layers are separately equal to $N_1 = f \times s_1 = g \times p^{2u_1}$ and $N_2 = f \times s_2 = g \times p^{u_1+u_2}$. This reveals an interesting fact that the run-size for each layer is not necessarily equal to a power of prime, but a multiple of prime power. On the other hand, different collections of candidate $\boldsymbol{C}_1$ matrices are considered in Step 1. This setup is naively determined by my limited experience in searching the required designs. Typically, the computational cost and searching time can be reduced significantly, if different collections of candidate $\boldsymbol{C}_1$ matrices are chosen for an odd g and an even g, respectively. However, there is currently no theoretical support for this setup. Fortunately, it performs quite well within the range of design parameters explored in the present study. In practice, the proposed algorithm is computationally infeasible for searching designs with large parameters, and some analytical methods are therefore developed for addressing this issue in the next section.

3.2 Construction Through Existing NOAs

Suppose that the triplet $(\boldsymbol{L}_{10}, \boldsymbol{L}_{20}, \phi)$ constitutes a $NOA((f_1, f_2), s_1, (s_1, s_2), 2)$, it could be obtained by implementing the proposed algorithm or other approaches in the literatures. Through an existing nested orthogonal array, an alternative one can be readily constructed as follows.

Method 1. Let $\boldsymbol{A}_1$ be an $(s_1 s_2) \times (s_1 s_2 + 1)$ matrix, and $\boldsymbol{C}_1$ be an $(s_1 s_2) \times f_1$ matrix, where

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{1}_{s_2} \otimes \boldsymbol{F}_1 & \boldsymbol{I}_{s_1 s_2} \end{bmatrix}; \quad \text{and} \quad \boldsymbol{C}_1 = \boldsymbol{L}_{10}^T \otimes \boldsymbol{1}_{s_2}. \tag{2}$$

Alternatively, let $\boldsymbol{A}_2$ be an $(s_1 s_2) \times (s_1 s_2 + 1)$ matrix, and $\boldsymbol{C}_2$ be an $(s_1 s_2) \times f_2$ matrix, where

$$\boldsymbol{A}_2 = \phi(\boldsymbol{A}_1); \quad \text{and} \quad \boldsymbol{C}_2 = \phi(\boldsymbol{L}_{20}^T) \otimes \boldsymbol{1}_{s_2}. \tag{3}$$

By collecting all the solutions of linear equations determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ separately given as (2) and (3) over Galois fields, a new nested orthogonal array can be obtained immediately.

When an s_1-symbol orthogonal array of strength two $\boldsymbol{L}$ is available, as mentioned earlier, then $\boldsymbol{L} \oplus \boldsymbol{D}$ and $\phi(\boldsymbol{L}) \oplus \phi(\boldsymbol{D})$ are s_1- and s_2-symbol orthogonal arrays of strength two, where $\boldsymbol{D}$ and $\phi(\boldsymbol{D})$ denote the $s_1 \times s_1$ difference matrices of s_1 and s_2 symbols, respectively. In the same spirit as Method 1, another construction method can then be devised as follows.

Method 2. Let $\boldsymbol{A}_1$ be an $(s_1^2 s_2) \times (s_1^2 s_2 + 1)$ matrix, and $\boldsymbol{C}_1$ be an $(s_1^2 s_2) \times (f_1 s_1)$ matrix, where

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{1}_{s_1 s_2} \otimes \boldsymbol{F}_1 & \boldsymbol{I}_{s_1^2 s_2} \end{bmatrix}; \quad \text{and} \quad \boldsymbol{C}_1 = \boldsymbol{L}_{10}^T \oplus \boldsymbol{D}^T \otimes \boldsymbol{1}_{s_2}. \tag{4}$$

In addition, let $\boldsymbol{A}_2$ be an $(s_1^2 s_2) \times (s_1^2 s_2 + 1)$ matrix, and $\boldsymbol{C}_2$ be an $(s_1^2 s_2) \times (f_2 s_1)$ matrix, where

$$\boldsymbol{A}_2 = \phi(\boldsymbol{A}_1); \quad \text{and} \quad \boldsymbol{C}_2 = \phi(\boldsymbol{L}_{20}^T) \oplus \phi(\boldsymbol{D})^T \otimes \boldsymbol{1}_{s_2}. \tag{5}$$

By solving the linear systems determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ separately given as (4) and (5) over Galois fields, another nested orthogonal array can be readily obtained. The design parameters of nested orthogonal arrays obtained by Methods 1 and 2 are summarized as follows.

Theorem 1. (a) The triplet $(\boldsymbol{L}_1, \boldsymbol{L}_2, \phi)$ comprises a $NOA((f_1 s_1, f_2 s_2), s_1 s_2 + 1, (s_1, s_2), 2)$, if $\boldsymbol{L}_1$ is determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ exhibited in (2), and $\phi(\boldsymbol{L}_2)$ is determined by $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ exhibited in (3). (b) The triplet $(\boldsymbol{L}_1, \boldsymbol{L}_2, \phi)$ constitutes a $NOA((f_1 s_1^2, f_2 s_1 s_2), s_1^2 s_2 + 1, (s_1, s_2), 2)$, if $\boldsymbol{L}_1$ is determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ exhibited in (4), and $\phi(\boldsymbol{L}_2)$ is determined by $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ exhibited in (5).

Because of the fact that the $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ matrices used in Methods 1 and 2 are constructed through an existing orthogonal array, the resulting $\boldsymbol{v}_1^T \boldsymbol{C}_1$ and $\boldsymbol{v}_2^T \boldsymbol{C}_2$ are found to be uniform on Galois fields for those $wt(\boldsymbol{v}_1^T \boldsymbol{A}_1) \leq 2$ and $wt(\boldsymbol{v}_2^T \boldsymbol{A}_2) \leq 2$, respectively. By the conditions described in Corollary 1, these matrix pairs can be employed for generating nested orthogonal arrays. The following examples are given for illustrating the applications of Methods 1 and 2.

Example 3. Suppose that the input-output relationship between the system output and nine input variables is of interest, and a $NOA((64, 16), 9, (4, 2), 2)$ is required for planning a series of computer simulations with two levels of accuracy. Let $\boldsymbol{A}_1$ be an 8×9 matrix, and $\boldsymbol{C}_1$ be an 8×16 matrix, separately given by

$$\boldsymbol{A}_1 = \left[\begin{array}{cc} \mathbf{1}_2 \otimes \boldsymbol{F}_1 & \boldsymbol{I}_8 \end{array} \right] \text{ and } \boldsymbol{C}_1 = \boldsymbol{L}_{10}^T \otimes \mathbf{1}_2,$$

where $\boldsymbol{F}_1$ represents the vector consisting of the four elements of $GF(4)$ in lexicographic order; and $\boldsymbol{L}_{10}$ denotes the projection onto the first four columns of $\boldsymbol{L}_1$ presented in (**??**). On the other hand, let $\boldsymbol{A}_2$ be an 8×9 matrix, and $\boldsymbol{C}_2$ be an 8×8 matrix, separately given by

$$\boldsymbol{A}_2 = \phi(\boldsymbol{A}_1) = \left[\begin{array}{cc} \mathbf{1}_4 \otimes \boldsymbol{F}_2 & \boldsymbol{I}_8 \end{array} \right] \text{ and } \boldsymbol{C}_2 = \phi(\boldsymbol{L}_{20}^T) \otimes \mathbf{1}_2,$$

where $\boldsymbol{F}_2$ stands for the vector consisting of the two elements of $GF(2)$ in lexicographic order; and $\phi(\boldsymbol{L}_{20})$ represents the projection onto the first four columns of $\phi(\boldsymbol{L}_2)$ presented in (**??**). The matrix pairs $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ satisfy the statement (a) of Theorem 1, the corresponding triplet $(\boldsymbol{L}_1, \boldsymbol{L}_2, \phi)$ constitutes a $NOA((64, 16), 9, (4, 2), 2)$. $\square$

Suppose that the matrix pairs $(\boldsymbol{A}_{10}, \boldsymbol{C}_{10})$ and $(\boldsymbol{A}_{20}, \boldsymbol{C}_{20})$ together with the Galois field projection ϕ determine a $NOA((f_0 s_1^{n_0-k_0}, f_0 s_2^{n_0-k_0}), n_0, (s_1, s_2), 2)$, where $\boldsymbol{A}_{10}$ is a $k_0 \times n_0$ matrix; $\boldsymbol{C}_{10}$ is a $k_0 \times f_0$ matrix; $\boldsymbol{A}_{20} = \phi(\boldsymbol{A}_{10})$; and $\boldsymbol{C}_{20} = \phi(\boldsymbol{C}_{10})$. Based on these elements, another nested orthogonal array can be derived through the following method.

Method 3. Let $\boldsymbol{A}_1$ be a $(k_0 s_1) \times (n_0 s_1)$ matrix, and $\boldsymbol{C}_1$ be a $(k_0 s_1) \times (f_0 s_1)$ matrix, where

$$\boldsymbol{A}_1 = \boldsymbol{A}_{10} \oplus \boldsymbol{D}^T; \text{ and } \boldsymbol{C}_1 = \boldsymbol{C}_{10} \oplus \boldsymbol{D}^T. \tag{6}$$

Alternatively, let $\boldsymbol{A}_2$ be a $(k_0 s_1) \times (n_0 s_1)$ matrix, and $\boldsymbol{C}_2$ be a $(k_0 s_1) \times (f_0 s_1)$ matrix, where

$$\boldsymbol{A}_2 = \phi(\boldsymbol{A}_1) = \boldsymbol{A}_{20} \oplus \phi(\boldsymbol{D})^T; \text{ and } \boldsymbol{C}_2 = \phi(\boldsymbol{C}_1) = \boldsymbol{C}_{20} \oplus \phi(\boldsymbol{D})^T. \tag{7}$$

By collecting all the solutions of linear equations determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ separately given as (6) and (7) over Galois fields, a new nested orthogonal array can be immediately derived, and its parameters are summarized as follows.

Theorem 2. The triplet $(\boldsymbol{L}_1, \boldsymbol{L}_2, \phi)$ comprises a $NOA((f_0 s_1^{(n_0-k_0)s_1+1}, f_0 s_1 s_2^{(n_0-k_0)s_1}), n_0 s_1, (s_1, s_2), 2)$, if $\boldsymbol{L}_1$ is determined by $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ exhibited in (6), and $\phi(\boldsymbol{L}_2)$ is determined by $(\boldsymbol{A}_2, \boldsymbol{C}_2)$ exhibited in (7).

Obviously, Method 3 tends to generate designs with large run-sizes. The row ranks of $\boldsymbol{A}_1$ in (6) and $\boldsymbol{A}_2$ in (7) are both equal to $k_0 s_1$, leading to that the run-sizes of single-flat designs are separately equal to $s_1^{(n_0-k_0)s_1}$ and $s_2^{(n_0-k_0)s_1}$ for the two layers.

3.3 Design Catalogue

For a given number of input variables n, computational and analytical methods are combined for constructing the proposed designs. Firstly, the proposed algorithm is used to generate designs with small to moderate run-sizes for $n = 5$ and $(s_1, s_2) = (4, 2)$. By implementing the proposed algorithm, a computer search is carried out for seeking designs with $f = 4$, 8, 12, 16, 20, 24 and 28, respectively. The obtained designs are named as Series 0, which not only serve as stepping stones for planning computer simulations, but also offer some templates for generating designs with large parameters. For a given design from Series 0, let $\boldsymbol{L}_{10}$ and $\phi(\boldsymbol{L}_{20})$ be the projections onto the first four columns of the two layers, respectively. Using Method 1 with $\boldsymbol{L}_{10}$ and $\phi(\boldsymbol{L}_{20})$, a new series of nested orthogonal arrays is constructed through (2) and (3) for $n = 9$, and named as Series 1. Similarly, using Method 2 with $\boldsymbol{L}_{10}$ and $\phi(\boldsymbol{L}_{20})$, an alternative collection of designs called Series 2 can be derived through (4) and (5) for $n = 33$.

On the other hand, let $(\boldsymbol{A}_{10}, \boldsymbol{C}_{10})$ be the matrix pair for determining a design from Series 0, 1 or 2. Using Method 3, three different collections of nested orthogonal arrays, which are separately named as Series 3, 4 and 5, can be generated through (6) and (7) for $n = 20$, 36 and 132, respectively. All the obtained designs are collected in the online supplementary materials available on the author's personal website (http://www.shinfu.idv.tw), and the design parameters are summarized in Table 1. Note that the supplementary materials include several lists of nested orthogonal arrays in terms of $(\boldsymbol{A}_1, \boldsymbol{C}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{C}_2)$. In particular, the designs of Series 0 are also available in terms of $(\boldsymbol{L}_1, \boldsymbol{L}_2)$.

Table 1: Parameters of designs in the catalogue

Series	n	(N_1, N_2)
0	5	(16, 8), (32, 16), (48, 24), (64, 32), (80, 40), (96, 48), (112, 56)
1	9	(64, 16), (128, 32), (192, 48), (256, 64), (320, 80), (384, 96), (448, 112)
2	33	(256, 64), (512, 128), (768, 192), (1024, 256), (1280, 320), (1536, 384), (1792, 448)
3	20	(4096, 256), (8192, 512), (12288, 768), (16384, 1024), (20480, 1280) (24576, 1536), (28672, 1792)
4	36	(16384, 1024), (32768, 2048), (49152, 3072), (65536, 4096), (81920, 5120) (98304, 6144), (114688, 7168)
5	132	(65536, 4096), (131072, 8192), (196608, 12288), (262144, 16384), (327680, 20480) (393216, 24576), (458752, 28672)

References

[1] Addelman, S. (1961). Irregular fractions of the 2^n factorial experiment. *Technometrics*, 3, 479-496.

[2] Ai, M., Jiang, B. and Li, K. (2014). Construction of sliced space-filling designs based on balanced sliced orthogonal arrays. *Statistica Sinica*, 24, 1685-1702.

[3] Butler, N. A. and Ramos, V. M. (2007). Optimal additions to and deletions from two- level orthogonal arrays. *Journal of the Royal Statistical Society: Series B*, 69, 51-61.

[4] Bose, R. C. and Bush, K. A. (1952). Orthogonal arrays of strength two and three. *Annals of Mathematical Statistics*, 23, 508-524.

[5] Cheng, C. S. (2014). Theory of factorial design: single- and multi-stratum experiments. New York: Chapman & Hall/CRC.

[6] Connor, W. S. and Young, S. (1961). Fractional factorial designs for experiments with factors at two and three levels (Applied Mathematics Series). National Bureau of Standards, U. S. Government Printing Office. Washington, D. C.

[7] Dey, A. (2010). Construction of nested orthogonal arrays. *Discrete Mathematics*, 310, 2831-2834.

[8] Fang, K. T., Li, R. Z. and Sujianto, A. (2006). Design and modeling for computer experiments. New York: Chapman & Hall/CRC.

[9] Hedayat, A. S., Sloane, N. J. A. and Stufken, J. (1999). Orthogonal arrays: theory and applications. New York: Springer.

[10] John, P. W. M. (1962). Three quarter replicates of 2^n designs. *Biometrics*, 18, 172-184.

[11] Qian, P. Z. G., Ai, M. and Wu, C. F. J. (2009a). Construction of nested space-filling designs. *Annals of Statistics*, 37, 3616-3643.

[12] Qian, P. Z. G., Tang, B. and Wu, C. F. J. (2009b). Nested space-filling designs for computer experiments with two levels of accuracy. *Statistica Sinica*, 19, 287-300.

[13] Qian, P. Z. G. and Wu, C. F. J. (2009). Sliced space-filling designs. *Biometrika*, 96, 945-956.

[14] Santner, T. J., Willams, B. J. and Notz, W. I. (2006). The design and analysis of computer experiments. New York: Springer.

[15] Srivastava, J. N. and Throop, D. (1990). Orthogonal arrays obtainable as solutions to linear equations over finite fields. *Linear Algebra and its Applications*, 127, 283-300.

[16] Sun, F., Yin, Y. and Liu, M. Q. (2013). Construction of nested space-filling designs using difference matrices. *Journal of Statistical Planning and Inference*, 143, 160-166.

[17] Sun, F., Liu, M. Q. and Qian, P. Z. G. (2014). On the construction of nested space-filling designs. *Annals of Statistics*, 42, 1394-1425.

[18] Tang, B. (1993). Orthogonal array-based Latin hypercubes. *Journal of the American Statistical Association*, 88, 1392-1397.

[19] Wu, C. F. J. and Hamada, M. (2009). Experiments: planning, analysis, and optimization. 2nd ed. New York: Wiley.

[20] Xie, H., Xiong, S., Qian, P. Z. G. and Wu, C. F. J. (2014). General sliced Latin hypercube designs. *Statistica Sinica*, 24, 1239-1256.

[21] Xu, X., Haaland, B. and Qian, P. Z. G. (2011). Sudoku-based space-filling designs. *Biometrika*, 98, 711-720.

[22] Yang, J. F., Lin, C. D., Qian, P. Z. G. and Lin, D. K. J. (2013). Construction of sliced orthogonal Latin hypercube designs. *Statistica Sinica*, 23, 1117-1130.

[23] Yin, Y., Lin, D. K. J. and Liu, M. Q. (2014). Sliced Latin hypercube designs via orthogonal arrays. *Journal of Statistical Planning and Inference*, 149, 162-171.