

A NEW AND FLEXIBLE DESIGN CONSTRUCTION FOR ORTHOGONAL ARRAYS FOR MODERN APPLICATIONS

BY YUANZHEN HE^{1,a} C. DEVON LIN^{2,b} AND FASHENG SUN^{3,c}

¹*School of Statistics, Beijing Normal University, heyuanzhen@bnu.edu.cn*

²*Department of Mathematics and Statistics, Queen's University, devon.lin@queensu.ca*

³*KLAS and School of Mathematics and Statistics, Northeast Normal University, sunfs359@nenu.edu.cn*

Orthogonal array, a classical and effective tool for collecting data, has been flourished with its applications in modern computer experiments and engineering statistics. Driven by the wide use of computer experiments with both qualitative and quantitative factors, multiple computer experiments, multifidelity computer experiments, cross-validation and stochastic optimization, orthogonal arrays with certain structures have been introduced. Sliced orthogonal arrays and nested orthogonal arrays are examples of such arrays. This article introduces a flexible, fresh construction method, which uses smaller arrays and a special structure. The method uncovers the hidden structure of many existing fixed-level orthogonal arrays of given run sizes, possibly with more columns. It also allows fixed-level orthogonal arrays of nearly strength three to be constructed, which are useful as there are not many construction methods for fixed-level orthogonal arrays of strength three, and also helpful for generating Latin hypercube designs with desirable low-dimensional projections. Theoretical properties of the proposed method are explored. As by-products, several theoretical results on orthogonal arrays are obtained.

1. Introduction. With the exponential growth of computing power, investigators are increasingly employing computer experiments, which stimulate real-world phenomena or complex systems using mathematical models and solving them using numerical methods such as computational fluid dynamics and finite element analysis, to help understand the respective systems. The underlying mechanisms of computer experiments are represented and implemented by computer codes [34, 35]. They frequently involve both qualitative and quantitative factors [4, 22, 33, 50]. For a given system, investigators are often faced with different computer codes because of the proliferation of mathematical models and numerical methods for their solution. As such, multiple computer experiments for the same system are performed [48]. Computer codes can be also executed at various degrees of accuracy, resulting in multifidelity computer experiments [7, 10, 42].

In contrast to the traditional physical experiments, greater numbers of input variables are involved and larger numbers of runs are employed in computer experiments. However, often only a few of input variables are thought to be of primary importance. To select important input variables, space-filling designs with desirable low-dimensional projection properties are commonly used. Such designs can be generated using orthogonal array based Latin hypercubes [29, 40]. This calls for orthogonal arrays of strength two, three or higher. To accommodate computer experiments with qualitative and quantitative factors and multiple computer experiments, sliced orthogonal array based Latin hypercubes were employed [12, 23, 24, 32]. Nested orthogonal array based Latin hypercubes were introduced to choose

Received December 2019; revised December 2021.

MSC2020 subject classifications. Primary 62K15; secondary 62K05.

Key words and phrases. Completely resolvable, difference scheme, fractional factorial design, computer experiment, nested orthogonal array, sliced orthogonal array.

inputs in multi-fidelity computer experiments [31]. The objective of this article is to propose a flexible method that allows orthogonal arrays, sliced orthogonal arrays and nested orthogonal arrays to be constructed.

To balance the run size economy and the higher dimensional projection property, we consider a class of designs, called *orthogonal arrays of nearly strength three*, which for a given run size, enjoy a higher dimensional projection property than orthogonal arrays of strength two and accommodate more factors than those of strength three [18, 19]. This class of designs is in the same spirit as *nearly orthogonal arrays* and can be viewed as their generalization [25, 41, 44]. They can be used to generate Latin hypercube designs with desirable low-dimensional projections which have been shown beneficial in several problems such as studying design optimality [39], and mean and variance estimation, which further are useful in numerical integration, uncertainty quantification and sensitivity analysis [11].

Research on orthogonal arrays has been a central part of design theory for decades and there are extensive and numerous results on their construction, optimality criteria and analysis. For excellent sources of reference for this topic see, for example, [5, 21, 28, 43, 46]. The applications of orthogonal arrays are pervasive in many areas such as computer experiments, integration, visualization, optimization and computer science [8, 20, 29]. The proposed method discovers a new design structure that allows orthogonal arrays of strength two, near strength three and exact strength three, resolvable orthogonal arrays, sliced orthogonal arrays and nested orthogonal arrays to be found. We provide *new* orthogonal arrays of strength three, sliced orthogonal arrays and nested orthogonal arrays.

It has been long since the concept of resolvable orthogonal arrays was introduced [2]. There are scarce results on the construction and the use of such arrays. Gupta et al. [9] was the first to discover the use of such arrays in producing orthogonal main-effects plans. Recently, it is shown by [49] and [27] that resolvable orthogonal arrays play a key role in constructing sliced Latin hypercubes and nearly orthogonal arrays that are mappable into fully orthogonal arrays, respectively. A completely resolvable orthogonal array is used for qualitative factors in marginally coupled designs for computer experiments with both qualitative and quantitative factors [3, 13, 14, 16, 47]. Notably, [37] and [21] are the only work that provided construction for resolvable orthogonal arrays with the former focusing on two-level arrays and the latter on multilevel arrays. The proposed method offers an alternative way to construct multilevel resolvable orthogonal arrays.

The remainder of this article is organized as follows. Section 2 provides necessary definitions, notation and background. Design construction and its use in constructing orthogonal arrays of strength two are given in Section 3. Section 4 presents the applications of the proposed method in constructing a number of classes of orthogonal arrays. Conclusion and discussion are given in Section 5. All the proofs are relegated to the [Appendix](#).

2. Definitions, notation and background. Consider designs of n runs with m factors of s levels, where $2 \leq s \leq n$. If for every $n \times t$ submatrix of the design, say \mathbf{D} , each of all possible level combinations appears equally often, design \mathbf{D} is called an *orthogonal array of strength t* [21]. We use $\text{OA}(n, m, s, t)$ to denote such a design. A column is said to be *balanced* if each level appears equally often. A pair of columns is said to be *orthogonal* if they form an orthogonal array of strength two. Three columns are said to be *3-orthogonal* if they form an orthogonal array of strength three. In addition, an $\text{OA}(n, m, s, 2)$ is called *saturated* if $n = m(s - 1) + 1$.

Three special classes of orthogonal arrays that are relevant in this article are resolvable orthogonal arrays, sliced orthogonal arrays and nested orthogonal arrays. An $\text{OA}(n, m, s, t)$ \mathbf{D} is called α -*resolvable* if it can be expressed as $\mathbf{D} = (\mathbf{D}_1^T, \dots, \mathbf{D}_{n/(\alpha s)}^T)^T$ such that each of $\mathbf{D}_1, \dots, \mathbf{D}_{n/(\alpha s)}$ is an $\text{OA}(\alpha s, m, s, 1)$ [21], for $t \geq 2$. Such an orthogonal array is denoted by

TABLE 1

An orthogonal array $\mathbf{D} = \text{OA}(16, 3, 4, 2)$ and the level-collapsing projection δ : \mathbf{D} is also a $\text{CROA}(16, 3, 4, 2)$, a $\text{BSOA}(16, 3, 4, 2; 4, 2)$ and an $\text{NOA}(16, 3, 4, 2; 4, 2)$

	\mathbf{D}				$\delta(\mathbf{D})$	
\mathbf{D}_1	0	0	0	0	0	0
	2	1	3	1	0	1
	1	3	2	0	1	1
	3	2	1	1	1	0
\mathbf{D}_2	2	2	2	1	1	1
	0	3	1	0	1	0
	3	1	0	1	0	0
	1	0	3	0	0	1
\mathbf{D}_3	1	1	1	0	0	0
	3	0	2	1	0	1
	0	2	3	0	1	1
	2	3	0	1	1	0
\mathbf{D}_4	3	3	3	1	1	1
	1	2	0	0	1	0
	2	0	1	1	0	0
	0	1	2	0	0	1

$\text{ROA}(n, m, s, t; \alpha)$. If $\alpha = 1$, the corresponding design is known as a *completely resolvable orthogonal array* and denoted by $\text{CROA}(n, m, s, t)$. For example, the orthogonal array $\mathbf{D} = \text{OA}(16, 3, 4, 2)$ in Table 1 is a $\text{CROA}(16, 3, 4, 2)$.

Next, we provide the definitions of both sliced orthogonal arrays and nested orthogonal arrays. Both definitions involve certain mapping, which we call *level-collapsing projection*. A mapping $\delta(\cdot)$ is called level-collapsing projection if the mapping is from a set S with s elements into its own subset and satisfies (a) S can be divided into s_0 parts S_1, \dots, S_{s_0} , with each part having s/s_0 elements, and (b) for any two elements $x \in S_i, y \in S_j$, $\delta(x) = \delta(y)$ for $i = j$ and $\delta(x) \neq \delta(y)$ otherwise. Table 1 provides an example of the level-collapsing projection $\delta(0) = \delta(1) = 0$ and $\delta(2) = \delta(3) = 1$. An $\text{OA}(n, m, s, 2)$ \mathbf{D} is called a *sliced orthogonal array* if its n rows can be partitioned into v subarrays $\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_v$ such that each \mathbf{D}_i becomes an $\text{OA}(n_0, m, s_0, 2)$ with $n_0 = n/v$ after the s levels in each column of \mathbf{D} are collapsed to s_0 levels according to some level-collapsing projection. We denote such an array $\text{SOA}(n, m, s, 2; v, s_0)$. Furthermore, if each column in each slice \mathbf{D}_i of a sliced orthogonal array is balanced, that is, having equal frequency of s levels, it is called *balanced sliced orthogonal array* [1]. We use $\text{BSOA}(n, m, s, 2; v, s_0)$ to denote such an array. An $\text{OA}(n, m, s, 2)$ \mathbf{D} is called *nested orthogonal array* if \mathbf{D} contains a subarray \mathbf{D}_0 , which becomes an $\text{OA}(n_0, m, s_0, 2)$ after level collapsing each column of \mathbf{D} according to certain level-collapsing projection [30]. We use $\text{NOA}(n, m, s, 2; n_0, s_0)$ to denote such an array. As an illustration, Table 1 presents a $\mathbf{D} = \text{BSOA}(16, 3, 4, 2; 4, 2)$, which is also an $\text{NOA}(16, 3, 4, 2; 4, 2)$ with any given \mathbf{D}_i being the subarray \mathbf{D}_0 , for $i = 1, 2, 3, 4$. A sliced orthogonal array is also a nested orthogonal array but the reverse does not hold.

A relevant concept for orthogonal arrays is difference scheme. An $r \times c$ array with entries from a finite Abelian group containing s entries is called a difference scheme if each vector difference between any two distinct columns of the array consists of every element from the finite Abelian group equally often [2]. Such an array is denoted by $\text{D}(r, c, s)$.

The proposed construction in Section 3 uses two key operators for matrices. Before reviewing the operators, we review the concept of Galois field. A field is called a Galois field (or

finite field) if it contains a finite number of elements and the operations of the addition, subtraction, multiplication and division on its elements satisfy the rules of arithmetic [21]. The order of a Galois field is defined to be the number of elements, and must be a prime power. Let $\text{GF}(s)$ denote the Galois field of order s , and throughout we let $\text{GF}(s) = \{\alpha_0, \alpha_1, \dots, \alpha_{s-1}\}$ with $\alpha_0 = 0$ and $\alpha_1 = 1$. If $s = p$ is a prime, $\text{GF}(p) = \{0, 1, \dots, p-1\}$. If $s = p^u$ for $u \geq 2$ and a prime p , $\text{GF}(p^u) = \{a_0 + a_1x + \dots + a_{u-1}x^{u-1} : a_0, \dots, a_{u-1} \in \text{GF}(p)\}$. That is, the elements of $\text{GF}(p^u)$ are the polynomials with the degree less than u and the coefficients from $\text{GF}(p)$. The addition of $\text{GF}(p^u)$ is the ordinary polynomial addition with the coefficients modulo p , and the multiplication is the ordinary polynomial multiplication and then modulo a given irreducible polynomial of degree u , where an irreducible polynomial is a polynomial that cannot be factored into the product of two nonconstant polynomials. For examples of the addition and multiplication of a Galois field, readers are referred to [21].

Let $\mathbf{A} = (a_{ij})$ be an $n_1 \times m_1$ matrix and \mathbf{B} be an $n_2 \times m_2$ matrix, where both matrices have entries from the Galois field $\text{GF}(s)$. Throughout, we let $\text{GF}(s) = \{\alpha_0, \alpha_1, \dots, \alpha_{s-1}\}$ with $\alpha_0 = 0$ and $\alpha_1 = 1$. The *Kronecker sum* of \mathbf{A} and \mathbf{B} is an $(n_1n_2) \times (m_1m_2)$ matrix given by

$$(1) \quad \mathbf{A} \oplus \mathbf{B} = [\mathbf{B}^{a_{ij}}]_{1 \leq i \leq n_1, 1 \leq j \leq m_1},$$

where $\mathbf{B}^{a_{ij}} = (\mathbf{B} + a_{ij})$ is an $n_2 \times m_2$ matrix with $+$ representing the addition in a field. If we partition the rows of \mathbf{B} into n_1 matrices, \mathbf{B}_i 's, and let \mathbf{a}_i be the i th row of \mathbf{A} , the *generalized Kronecker sum* of \mathbf{A} and \mathbf{B} is defined as

$$(2) \quad \mathbf{A} \circledast \mathbf{B} = [\mathbf{a}_i \oplus \mathbf{B}_i]_{1 \leq i \leq n_1} = \begin{pmatrix} \mathbf{a}_1 \oplus \mathbf{B}_1 \\ \vdots \\ \mathbf{a}_{n_1} \oplus \mathbf{B}_{n_1} \end{pmatrix},$$

where the operator \oplus is given in (1). Note that the number of runs in $\mathbf{A} \circledast \mathbf{B}$ is the same as that of \mathbf{B} and the number of factors in $\mathbf{A} \circledast \mathbf{B}$ is m_1m_2 , the product of the number of factors in \mathbf{A} and the number of factors in \mathbf{B} . For example, let $\mathbf{A} = (0, 1, 2)^T$ and

$$\mathbf{B} = \begin{pmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \\ \mathbf{B}_3 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 2 & 2 & 1 \\ - & - & - & - \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 2 & 0 \\ 1 & 2 & 0 & 2 \\ - & - & - & - \\ 2 & 0 & 2 & 2 \\ 2 & 1 & 0 & 1 \\ 2 & 2 & 1 & 0 \end{pmatrix}, \quad \text{then we have } \mathbf{A} \circledast \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 2 & 2 & 1 \\ 2 & 1 & 2 & 2 \\ 2 & 2 & 0 & 1 \\ 2 & 0 & 1 & 0 \\ 1 & 2 & 1 & 1 \\ 1 & 0 & 2 & 0 \\ 1 & 1 & 0 & 2 \end{pmatrix}.$$

3. Design construction and preliminary results. This section introduces a new construction and presents some preliminary results for the later development. The proposed construction uses an $n_1 \times m_1$ matrix \mathbf{A} and n_1 matrices $\mathbf{B}_1, \dots, \mathbf{B}_{n_1}$ each of size $n_2 \times m_2$, where the entries of \mathbf{A} and \mathbf{B}_i 's are from the Galois field $\text{GF}(s) = \{\alpha_0, \alpha_1, \dots, \alpha_{s-1}\}$. Assume s is a prime power throughout. Denote the rows of \mathbf{A} by $\mathbf{a}_1, \dots, \mathbf{a}_{n_1}$, and stack $\mathbf{B}_1, \dots, \mathbf{B}_{n_1}$ row by row and denote the resulting matrix by $\mathbf{B} = (\mathbf{B}_1^T, \dots, \mathbf{B}_{n_1}^T)^T$. We define $s+1$ arrays $\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_{s+1}$ as follows. For $\alpha_g \in \text{GF}(s)$ and $g = 1, \dots, s-1$, define

$$(3) \quad \mathbf{D}_g = \mathbf{A} \circledast (\alpha_g * \mathbf{B}) = \begin{bmatrix} \mathbf{a}_1 \oplus (\alpha_g * \mathbf{B}_1) \\ \vdots \\ \mathbf{a}_{n_1} \oplus (\alpha_g * \mathbf{B}_{n_1}) \end{bmatrix},$$

and

$$(4) \quad \mathbf{D}_s = \mathbf{0}_{n_1} \circledast \mathbf{B} = \begin{bmatrix} 0 \oplus \mathbf{B}_1 \\ \vdots \\ 0 \oplus \mathbf{B}_{n_1} \end{bmatrix},$$

as well as

$$(5) \quad \mathbf{D}_{s+1} = \mathbf{A} \oplus \mathbf{0}_{n_2},$$

where \circledast in (3) represents the multiplication in a field and $\mathbf{0}_{n_2}$ denotes a column vector with n_2 zeros. Now obtain an array

$$(6) \quad \mathbf{E} = [\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_{s+1}],$$

where each \mathbf{D}_g is an $(n_1 n_2) \times (m_1 m_2)$ array by the definition of the generalized Kronecker sum, $g = 1, 2, \dots, s-1$, \mathbf{D}_s is an $(n_1 n_2) \times m_2$ array and \mathbf{D}_{s+1} is an $(n_1 n_2) \times m_1$ array, and thus \mathbf{E} is an $(n_1 n_2) \times [(s-1)m_1 m_2 + m_1 + m_2]$ array.

To the best of our knowledge, the construction of \mathbf{E} in (6) is a new way of constructing designs. The construction is flexible as it can be used to construct different types of orthogonal arrays with different choices of \mathbf{A} and \mathbf{B}_i 's. It produces a rich, vast class of designs with non-isomorphic \mathbf{A} 's, nonisomorphic \mathbf{B}_i 's, as well as isomorphic \mathbf{B}_i 's up to column permutations and/or level relabeling of one or more columns (e.g., [45] demonstrates that the level relabeling of columns improves the design properties). In Section 4.1, we illustrate this strategy to provide new orthogonal arrays of strength three.

To prepare for the detailed and quantitative illustration of the use of the proposed construction, we provide some preliminary results summarized in Lemmas 3.1 and 3.2. Some notation are in order. Let \mathbf{a} , \mathbf{c} and \mathbf{e} be a column vector of length n_1 , respectively, whose entries are from $\text{GF}(s)$. For $i = 1, \dots, n_1$, let \mathbf{b}_i , \mathbf{d}_i and \mathbf{f}_i be a column vector of length n_2 . The entries of \mathbf{b}_i 's, \mathbf{d}_i 's and \mathbf{f}_i 's are also from $\text{GF}(s)$. Stacking all \mathbf{b}_i 's row by row, we obtain $\mathbf{b} = (\mathbf{b}_1^T, \dots, \mathbf{b}_{n_1}^T)^T$. Similarly, obtain $\mathbf{d} = (\mathbf{d}_1^T, \dots, \mathbf{d}_{n_1}^T)^T$ and $\mathbf{f} = (\mathbf{f}_1^T, \dots, \mathbf{f}_{n_1}^T)^T$. Lemma 3.1 provides the sufficient conditions for two columns to be orthogonal when one column is obtained by the generalized Kronecker sum of \mathbf{a} and \mathbf{b} and the other column is obtained by the generalized Kronecker sum of \mathbf{c} and \mathbf{d} .

LEMMA 3.1. *Given $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ defined above, the array $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \mathbf{d})$ is an $OA(n_1 n_2, 2, s, 2)$ if one of the following conditions is satisfied, for $i = 1, \dots, n_1$ and $\alpha \in \text{GF}(s)$:*

- (i) $(\mathbf{b}_i, \mathbf{d}_i)$ is an $OA(n_2, 2, s, 2)$;
- (ii) \mathbf{b}_i is balanced, $\mathbf{d} = \alpha \mathbf{b}$, and one of the following four conditions is satisfied: (1) $\alpha = 1$ and (\mathbf{a}, \mathbf{c}) is a $D(n_1, 2, s)$; (2) (\mathbf{a}, \mathbf{c}) is an $OA(n_1, 2, s, 2)$; (3) $\mathbf{c} = \mathbf{0}$, $\alpha \neq 0$, and \mathbf{a} is balanced and (4) $\mathbf{a} = \mathbf{c}$, $\alpha \neq 1$, and \mathbf{a} is balanced; and
- (iii) $\mathbf{d} = \mathbf{0}$, and one of the following two conditions is satisfied: (1) \mathbf{b}_i and \mathbf{c} are balanced; and (2) $\mathbf{b} = \mathbf{0}$ and (\mathbf{a}, \mathbf{c}) is an $OA(n_1, 2, s, 2)$.

Lemma 3.2 below provides the sufficient conditions for three columns each obtained by the generalized Kronecker sum to be 3-orthogonal.

LEMMA 3.2. *Given $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}$ defined above, the array $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \mathbf{d}, \mathbf{e} \circledast \mathbf{f})$ is an $OA(n_1 n_2, 3, s, 3)$ if one of the following conditions is satisfied, for $i = 1, \dots, n_1$ and $\alpha, \beta \in \text{GF}(s)$:*

- (i) $(\mathbf{b}_i, \mathbf{d}_i, \mathbf{f}_i)$ is an $OA(n_2, 3, s, 3)$;
- (ii) $(\mathbf{b}_i, \mathbf{d}_i) = OA(n_2, 2, s, 2)$, $\mathbf{f} = \alpha \mathbf{b}$, and $(\mathbf{e}, \alpha \mathbf{a}) = D(n_1, 2, s)$;

- (iii) $\mathbf{d} = \beta \mathbf{b}$, $\mathbf{f} = \alpha \mathbf{b}$, \mathbf{b}_i is balanced, $(\mathbf{a}, \mathbf{c}) = \text{OA}(n_1, 2, s, 2)$ and one of the following three conditions is satisfied: (1) $\mathbf{e} = \mathbf{a}$ and $\alpha \neq 1$; or (2) $\mathbf{e} = \mathbf{0}$ and $\alpha \neq 0$ and (3) $(\mathbf{a}, \mathbf{c}, \mathbf{e}) = \text{OA}(n_1, 3, s, 3)$; and
- (iv) $\mathbf{b} = \mathbf{d} = \mathbf{0}$, \mathbf{f}_i is balanced, and $(\mathbf{a}, \mathbf{c}) = \text{OA}(n_1, 2, s, 2)$.

We now present one key result of the proposed construction (6). That is, for a prime power s , if \mathbf{A} and \mathbf{B}_i 's in (6) are s -level orthogonal arrays of strength two, so is the resulting design \mathbf{E} .

THEOREM 3.1. *If $\mathbf{A} = \text{OA}(n_1, m_1, s, 1)$ for $m_1 = 1$ or $\text{OA}(n_1, m_1, s, 2)$ for $m_1 > 1$, and $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 2)$, for $i = 1, \dots, n_1$ and a prime power s , then design \mathbf{E} in (6) is an $\text{OA}(n_1 n_2, (s - 1)m_1 m_2 + m_1 + m_2, s, 2)$.*

Theorem 3.1 indicates that an s -level orthogonal array of strength two can be constructed using only s -level orthogonal arrays of strength two with smaller sizes. To the best of our knowledge, such a construction is fundamentally different from the existing methods such as direct constructions or those using difference schemes and smaller orthogonal arrays [21]. To help understand Theorem 3.1, one may think the columns of \mathbf{E} are derived in the following way. Arrays \mathbf{A} and \mathbf{B} accommodate m_1 and m_2 factors, respectively. The columns of \mathbf{D}_{s+1} in (5) and those of \mathbf{D}_s in (4) can be thought of being derived from the main effects of factors in \mathbf{A} and the main effects of factors in \mathbf{B} , respectively. The columns of $\mathbf{D}_1, \dots, \mathbf{D}_{s-1}$ can be viewed as representing the interaction effects of the factors in \mathbf{A} and those in \mathbf{B} . For example, for $s = 2$, \mathbf{D}_1 constitutes the two-factor interactions between the factors in \mathbf{A} and those in \mathbf{B} ; for $s = 3$, $\mathbf{D}_1 = \mathbf{A} \circledast \mathbf{B}$ and $\mathbf{D}_2 = \mathbf{A} \circledast (2 * \mathbf{B})$ amount to the linear-by-linear interactions and the linear-by-quadratic interactions, respectively, which together constitute the interaction effects between \mathbf{A} and \mathbf{B} .

Example 1 is an illustration of the use of Theorem 3.1.

EXAMPLE 1. Consider constructing three-level orthogonal arrays of 81 runs and strength 2. It is well known that an $\text{OA}(81, 40, 3, 2)$ can be constructed via the Rao–Hamming construction [21]. The proposed construction (6) can also provide $\text{OA}(81, 40, 3, 2)$'s using two different ways: (1) using $\mathbf{A} = \text{OA}(3, 1, 3, 1)$ and $\mathbf{B}_i = \text{OA}(27, 13, 3, 2)$ for $i = 1, 2, 3$, where \mathbf{B}_i can be any design that is isomorphic to any of the 68 nonisomorphic $\text{OA}(27, 13, 3, 2)$'s given on the website [6]; and (2) using $\mathbf{A} = \text{OA}(9, 4, 3, 2)$ and $\mathbf{B}_i = \text{OA}(9, 4, 3, 2)$ for $i = 1, \dots, 9$, where \mathbf{A} and \mathbf{B}_i 's can be any design that is isomorphic to the unique $\text{OA}(9, 4, 3, 2)$ given on the website [36].

Next, we apply Theorem 3.1 to obtain some specific classes of orthogonal arrays. Corollaries 3.1–3.3 summarize that the proposed construction can provide two-level *saturated* orthogonal arrays whose run sizes are multiples of 8, s -level *saturated* orthogonal arrays of s^u runs and s -level orthogonal arrays of $2s^u$ runs, where $u \geq 2$ is an integer and s is a prime power. It shall be noted that these three corollaries also hold for a trivial case that \mathbf{A} is a column vector $(\alpha_0, \dots, \alpha_{s-1})^T$ for obtaining s -level orthogonal arrays.

COROLLARY 3.1. *If $\mathbf{A} = \text{OA}(n_1, n_1 - 1, 2, 2)$ and $\mathbf{B}_i = \text{OA}(n_2, n_2 - 1, 2, 2)$, for $i = 1, \dots, n_1$, then design \mathbf{E} in (6) is an $\text{OA}(n_1 n_2, n_1 n_2 - 1, 2, 2)$.*

COROLLARY 3.2. *If $\mathbf{A} = \text{OA}(s^{k_1}, (s^{k_1} - 1)/(s - 1), s, 2)$ and $\mathbf{B}_i = \text{OA}(s^{k_2}, (s^{k_2} - 1)/(s - 1), s, 2)$, for $i = 1, \dots, s^{k_1}$ and a prime power s , then design \mathbf{E} in (6) is an $\text{OA}(s^{k_1+k_2}, (s^{k_1+k_2} - 1)/(s - 1), s, 2)$.*

COROLLARY 3.3. *If $\mathbf{A} = \text{OA}(s^{k_1}, (s^{k_1} - 1)/(s - 1), s, 2)$ and $\mathbf{B}_i = \text{OA}(2s^{k_2}, 2(s^{k_2} - 1)/(s - 1) - 1, s, 2)$, for $i = 1, \dots, s^{k_1}$ and a prime power $s > 2$, then design \mathbf{E} in (6) is an $\text{OA}(2s^{k_1+k_2}, 2(s^{k_1+k_2} - 1)/(s - 1) - 1 - s(s^{k_1} - 1)/(s - 1), s, 2)$.*

REMARK 3.1. The family of $\text{OA}(2s^k, 2(s^k - 1)/(s - 1) - 1, s, 2)$ can be obtained via the use of difference schemes for a prime power s and $k \geq 2$ (Theorem 6.40 of [21]). Corollary 3.3 reveals that the construction (6) provides $s(s^{k_1} - 1)/(s - 1)$ fewer columns than the difference schemes method. For a fixed $k = k_1 + k_2$, to maximize the number of columns in \mathbf{E} , $k_1 = 1$ should be chosen.

Table 2 provides examples of orthogonal arrays of strength two constructed via (6). The list is not meant to be exhaustive. Table 2 also lists the maximal number m^* of columns in an orthogonal array that is obtained by the approaches in the literature. It reveals that in all cases except run sizes being $2s^k$ with a prime power s , the proposed construction (6) provides orthogonal arrays with the same m^* . We provide some results on the maximal number of columns in \mathbf{E} in (6) for given values of n , s and t in the Supplementary Material [15].

4. Applications. The versatility of the proposed construction is demonstrated in this section by showing that it can be used to construct many orthogonal arrays including those of strength three, resolvable orthogonal arrays, balanced sliced orthogonal arrays and nested orthogonal arrays. New orthogonal arrays of strength three, balanced sliced orthogonal arrays and nested orthogonal arrays are obtained. The basic idea is to couple certain types of orthogonal arrays with orthogonal arrays of strength two to generate orthogonal arrays of the same types but with larger run sizes and higher dimensionality. Another key result in this section is, by removing some columns from orthogonal arrays of strength two in the previous section, we get designs that have a very large proportion of 3-orthogonal columns while the numbers of their columns are much larger than the corresponding orthogonal arrays of strength three.

4.1. Construction of orthogonal arrays of strength three. In (3) and (4), we let $\mathbf{A} = \text{OA}(s, 1, s, 1)$ or $\mathbf{A} = \text{OA}(s^2, 2, s, 2)$ and \mathbf{B}_i 's be orthogonal arrays of strength three. Propositions 4.1 and 4.2 summarize the detailed results on the use of the proposed construction for orthogonal arrays of strength three. For a given run size and number of column in an orthogonal array of strength three, there are two different ways of construction via the proposed method. For example, an $\text{OA}(81, 8, 3, 3)$ can be obtained using either $\mathbf{A} = (0, 1, 2)^T$, $\mathbf{B}_i = \text{OA}(27, 4, 3, 3)$ for $i = 1, 2, 3$ in Proposition 4.1 or $\mathbf{A} = \text{OA}(9, 2, 3, 2)$, $\mathbf{B}_i = \text{OA}(9, 2, 3, 2)$ for $i = 1, \dots, 9$ in Proposition 4.2.

PROPOSITION 4.1. *If for $i = 1, \dots, s$ and a prime power s , \mathbf{A} is an $\text{OA}(s, 1, s, 1)$ and \mathbf{B}_i is an $\text{OA}(n_2, m_2, s, 2)$ for $m_2 = 2$ or an $\text{OA}(n_2, m_2, s, 3)$ for $m_2 \geq 3$, then for $1 \leq g \neq g' \leq s$, $(\mathbf{D}_g, \mathbf{D}_{g'})$ is an $\text{OA}(sn_2, 2m_2, s, 3)$, where \mathbf{D}_g is defined in (3) for $g = 1, \dots, s - 1$ and \mathbf{D}_s is defined in (4).*

PROPOSITION 4.2. *If for $i = 1, \dots, s^2$ and a prime power s , \mathbf{A} is an $\text{OA}(s^2, 2, s, 2)$, and \mathbf{B}_i is an $\text{OA}(n_2, m_2, s, 2)$ for $m_2 = 2$ or an $\text{OA}(n_2, m_2, s, 3)$ for $m_2 \geq 3$, then:*

- (i) *for $1 \leq g \neq g' \leq s - 1$, $(\mathbf{D}_g, \mathbf{D}_{g'})$ is an $\text{OA}(s^2n_2, 4m_2, s, 3)$, and*
- (ii) *for $1 \leq g \leq s - 1$, $(\mathbf{D}_g, \mathbf{D}_s)$ is an $\text{OA}(s^2n_2, 3m_2, s, 3)$,*

where \mathbf{D}_g is defined in (3) for $g = 1, \dots, s - 1$, and \mathbf{D}_s is defined in (4).

TABLE 2
Examples of orthogonal arrays of strength two obtained by (6)

s	n	m	n_1	m_1	n_2	m_2	m^*	Method
2	8	7	2	1	4	3	7	Corollaries 3.1 and 3.2
2	16	15	2	1	8	7	15	Corollaries 3.1 and 3.2
2	16	15	4	3	4	3	15	Corollaries 3.1 and 3.2
2	24	23	2	1	12	11	23	Corollary 3.1
2	32	31	2	1	16	15	31	Corollaries 3.1 and 3.2
2	32	31	4	3	8	7	31	Corollaries 3.1 and 3.2
2	40	39	2	1	20	19	39	Corollary 3.1
2	48	47	2	1	24	23	47	Corollary 3.1
2	48	47	4	3	12	11	47	Corollary 3.1
2	56	55	2	1	28	27	55	Corollary 3.1
2	64	63	2	1	32	31	63	Corollaries 3.1 and 3.2
2	64	63	4	3	16	15	63	Corollaries 3.1 and 3.2
2	64	63	8	7	8	7	63	Corollary 3.1
2	72	71	2	1	36	35	71	Corollary 3.1
2	80	79	2	1	40	39	79	Corollary 3.1
2	80	79	4	3	20	19	79	Corollary 3.1
2	88	87	2	1	44	43	87	Corollary 3.1
2	96	95	2	1	48	47	95	Corollary 3.1
2	96	95	4	3	24	23	95	Corollary 3.1
3	27	13	3	1	9	4	13	Corollary 3.2
3	54	22	3	1	18	7	25	Corollary 3.3
3	81	40	3	1	27	13	40	Corollary 3.2
3	81	40	9	4	9	4	40	Corollary 3.2
3	162	76	3	1	54	25	79	Corollary 3.3
4	64	21	4	1	16	5	21	Corollary 3.2
4	128	37	4	1	32	9	41	Corollary 3.3
4	256	85	4	1	64	21	85	Corollary 3.2
4	256	85	16	5	16	5	85	Corollary 3.2
5	125	31	5	1	25	6	31	Corollary 3.2
5	250	56	5	1	50	11	61	Corollary 3.3
5	625	156	5	1	125	31	156	Corollary 3.2
5	625	156	25	6	25	6	156	Corollary 3.2
7	343	57	7	1	49	8	57	Corollary 3.2
7	686	106	7	1	98	15	113	Corollary 3.3
7	2401	400	7	1	343	57	400	Corollary 3.2
7	2401	400	49	8	49	8	400	Corollary 3.2
8	512	73	8	1	64	9	73	Corollary 3.2
8	1024	137	8	1	128	17	145	Corollary 3.3
8	4096	585	8	1	512	73	585	Corollary 3.2
8	4096	585	64	9	64	9	585	Corollary 3.2
9	729	91	9	1	81	10	91	Corollary 3.2
9	1458	172	9	1	162	19	181	Corollary 3.3
9	6561	820	9	1	729	91	820	Corollary 3.2
9	6561	820	81	10	81	10	820	Corollary 3.2

Proposition 4.1 follows by parts (i) and (ii) of Lemma 3.2. Proposition 4.2 requires conditions (i), (ii), (iii.1) and (iii.2) in Lemma 3.2. Both propositions are powerful results in that by repeated applications of these results, many infinite series of orthogonal arrays of strength three can be obtained. For example, by Proposition 4.1, using an $OA(81, 10, 3, 3)$ for \mathbf{B}_i yields $OA(243, 20, 3, 3)$'s which in turn can be used to obtain a pool of $OA(729, 40, 3, 3)$'s and so on. The general result is given in Theorem 4.1. In addition, we derive some new results

TABLE 3
Examples of column permutations of \mathbf{B}_0 to obtain $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$ for nonisomorphic $\text{OA}(243, 20, 3, 3)$'s

Design 1		Design 2	
\mathbf{B}_1	2 10 4 5 3 8 7 1 6 9	\mathbf{B}_1	8 9 2 10 3 7 4 1 5 6
\mathbf{B}_2	5 2 1 7 6 8 9 10 3 4	\mathbf{B}_2	7 4 8 2 6 10 9 3 1 5
\mathbf{B}_3	5 4 10 1 8 6 9 3 2 7	\mathbf{B}_3	6 1 9 4 10 3 8 7 2 5
Design 3		Design 4	
\mathbf{B}_1	9 2 1 6 10 8 3 7 4 5	\mathbf{B}_1	3 7 1 9 8 4 10 5 6 2
\mathbf{B}_2	1 5 7 3 4 6 8 9 10 2	\mathbf{B}_2	6 10 5 3 4 1 9 8 7 2
\mathbf{B}_3	9 2 5 3 10 1 7 6 4 8	\mathbf{B}_3	6 8 9 7 10 3 5 4 2 1

on lower bounds of the maximal number of columns in certain orthogonal arrays of strength three in the Supplementary Material [15].

THEOREM 4.1. *If there exists an $\text{OA}(n_2, m_2, s, 3)$ for a prime power s , then $\text{OA}(n_2 s^k, 2^k m_2, s, 3)$'s for $s \geq 2$ can be constructed.*

Examples of the application of Theorem 4.1 are obtaining orthogonal arrays of strength three including an $\text{OA}(2^k, 2^{k-1}, 2, 3)$, an $\text{OA}(3 \cdot 2^k, 3 \cdot 2^{k-1}, 2, 3)$, an $\text{OA}(2 \cdot 3^k, 5 \cdot 2^{k-3}, 3, 3)$, an $\text{OA}(3^{k+1}, 10 \cdot 2^{k-3}, 3, 3)$ and an $\text{OA}(4^{k+1}, 17 \cdot 2^{k-3}, 4, 3)$, for $k \geq 3$, with small orthogonal arrays such as an $\text{OA}(8, 4, 2, 3)$, an $\text{OA}(24, 12, 2, 3)$, an $\text{OA}(54, 5, 3, 3)$, an $\text{OA}(81, 10, 3, 3)$ and an $\text{OA}(256, 17, 4, 3)$. Next, we give a detailed example of using the proposed construction to provide new orthogonal arrays of strength three in Example 2.

EXAMPLE 2. Consider constructing $\text{OA}(243, 20, 3, 3)$'s for which the only available one is provided in [36]. We show how to use Proposition 4.1 to construct new $\text{OA}(243, 20, 3, 3)$'s using an $\text{OA}(81, 10, 3, 3)$ which is unique and available at [6]. Let \mathbf{B}_0 be the $\text{OA}(81, 10, 3, 3)$. In Proposition 4.1, let $\mathbf{A} = (0, 1, 2)^T$, $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$ be obtained from \mathbf{B}_0 by column permutations, and the resulting design $\mathbf{D} = (\mathbf{D}_1, \mathbf{D}_2)$. By using different column permutations for $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$, we obtain many different \mathbf{D} 's some of which are isomorphic while some are non-isomorphic to each other and to the $\text{OA}(243, 20, 3, 3)$ in [36]. To save the space, we only list a few ways of obtaining nonisomorphic $\text{OA}(243, 20, 3, 3)$'s in Table 3 with the elements being the column permutation of \mathbf{B}_0 for the respective $\mathbf{B}_1, \mathbf{B}_2$ and \mathbf{B}_3 .

4.2. Construction of orthogonal arrays of near strength three. We introduce orthogonal arrays of near strength three. These designs are not of exact strength three, but of strength two. They have the feature that a large proportion of all possible three-column combinations is 3-orthogonal (see the definition of 3-orthogonality in Section 2). By slightly relaxing the exact strength-three orthogonality, these designs are shown to accommodate significantly more columns than orthogonal arrays of strength three with the same run sizes. To measure how close these designs are to orthogonal arrays of strength three in terms of 3-orthogonality, we define the following quantity for an orthogonal array \mathbf{D} with m columns:

$$(7) \quad p(\mathbf{D}) = \frac{r}{\binom{m}{3}},$$

where r is the number of three distinct columns that are 3-orthogonal. Note that $0 \leq p \leq 1$. An orthogonal array of strength three has $p = 1$. A larger value of p is preferred.

PROPOSITION 4.3. If $\mathbf{A} = \text{OA}(s, 1, s, 1)$, $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 3)$, for $i = 1, \dots, s$ and a prime power s , and \mathbf{F} is obtained by excluding the last column of \mathbf{E} in (6), then \mathbf{F} is an $\text{OA}(n_2 s, sm_2, s, 2)$ and

$$(8) \quad p(\mathbf{F}) = 1 - \frac{m_2 \binom{s}{3}}{\binom{sm_2}{3}} = 1 - \frac{(s-1)(s-2)}{(sm_2-1)(sm_2-2)}.$$

Proposition 4.3 can be easily verified by noting that $\mathbf{F} = (\mathbf{D}_1, \dots, \mathbf{D}_s)$ with \mathbf{D}_g in (3) for $g = 1, \dots, s-1$ and \mathbf{D}_s as in (4), and that any three columns that are not 3-orthogonal satisfy that each of them is from one of $\mathbf{D}_i, \mathbf{D}_j, \mathbf{D}_k$ with distinct values of i, j, k and obtained by the generalized Kronecker sum of the same column of \mathbf{B} . The last column of \mathbf{E} is removed to increase the value of p . It shall be noted that $p(\mathbf{F}) = 1$ when $s = 2$. In addition, it is straightforward to show that $p(\mathbf{F})$ in (8) is greater than $1 - 1/m_2^2$, and thus for a large value of m_2 , $p(\mathbf{F})$ is very close to 1. For example, $m_2 = 11$, $p(\mathbf{F})$ is greater than 0.991. Example 3 illustrates the use of Proposition 4.3 and indicates $p(\mathbf{F})$ can be very close to 1.

EXAMPLE 3. Consider $s = 3$, $\mathbf{B}_i = \text{OA}(81, 10, 3, 3)$, and thus $m_2 = 10$, the resulting \mathbf{F} is an $\text{OA}(243, 30, 3, 2)$ with $p(\mathbf{F}) = 0.998$ while a three-level orthogonal array of strength three with 243 runs can accommodate up to 20 columns (Table 12.2 of [21]). Consider $s = 4$, $\mathbf{B}_i = \text{OA}(64, 6, 4, 3)$, and thus $m_2 = 6$, the resulting \mathbf{F} is an $\text{OA}(256, 24, 4, 2)$ with $p(\mathbf{F}) = 0.988$ while a four-level orthogonal array of strength three with 256 runs can accommodate up to 17 columns (Table 12.6 of [21]).

PROPOSITION 4.4. If $\mathbf{A} = \text{OA}(s^2, 2, s, 2)$, $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 3)$, for $i = 1, \dots, s^2$ and a prime power s , and \mathbf{F} is obtained by excluding the last two columns of \mathbf{E} in (6), then \mathbf{F} is an $\text{OA}(n_2 s^2, (2s-1)m_2, s, 2)$ and

$$(9) \quad p(\mathbf{F}) = 1 - \frac{2m_2 \binom{s}{3}}{\binom{(2s-1)m_2}{3}} = 1 - \frac{2s(s-1)(s-2)}{(2s-1)(\gamma-1)(\gamma-2)},$$

where $\gamma = (2s-1)m_2$.

Proposition 4.4 can be shown using the same arguments as those for Proposition 4.3, with the help of Proposition 4.2. Noted that $p(\mathbf{F}) = 1$ when $s = 2$. For $s \geq 3$, it can be shown that

$$(10) \quad p(\mathbf{F}) = \begin{cases} 1 - \frac{2m_2[\binom{s-1}{3} + \binom{s-1}{2}]}{\binom{(2s-1)m_2+m_2}{3}}, & s > 3; \\ 1 - \frac{2m_2 \binom{s-1}{2}}{\binom{(2s-1)m_2+m_2}{3}}, & s = 3. \end{cases}$$

Thus, $p(\mathbf{F})$ in (9) can be very close to 1 with a large value of m_2 . As an illustration, for $s = 4$, $m_2 = 6$, $\mathbf{B}_i = \text{OA}(64, 6, 4, 3)$, the resulting \mathbf{F} of 1024 runs has 42 columns and $p(\mathbf{F}) = 0.997$ while the available four-level orthogonal array of 1024 runs can have up to 32 columns (Table 12.3 of [21]).

4.3. *Resolvable orthogonal arrays.* Proposition 4.5 states that pairing an orthogonal array and resolvable orthogonal arrays, the proposed method constructs resolvable orthogonal arrays of larger sizes.

PROPOSITION 4.5. If $\mathbf{A} = \text{OA}(n_1, m_1, s, 1)$ for $m_1 = 1$ or $\text{OA}(n_1, m_1, s, 2)$ for $m_1 > 1$, $\mathbf{B}_i = \text{ROA}(n_2, m_2, s, 2; \alpha)$ for $i = 1, \dots, n_1$, and a prime power s , and \mathbf{F} is obtained by excluding the last m_1 columns of \mathbf{E} in (6), then \mathbf{F} is a $\text{ROA}(n_1 n_2, (s-1)m_1 m_2 + m_2, s, 2; \alpha)$.

The last m_1 columns of \mathbf{E} are excluded such that the resulting \mathbf{F} is the generalized Kronecker sum of $(\mathbf{A}, \mathbf{0}_{n_1})$ and the resolvable orthogonal arrays \mathbf{B}_i 's, and thus it is also resolvable. The orthogonality of \mathbf{F} follows by Theorem 3.1. Proposition 4.5 offers a new way to construct resolvable orthogonal arrays, which are previously obtained via the difference schemes method [21]. In addition, the result implies an upper bound for the number of columns in a resolvable orthogonal array. To provide the bound, we first review a result, given by [37], in Lemma 4.1, which says the upper bound for the number of columns in an $n/(sr)$ -resolvable s -level orthogonal array is $(n - r)/(s - 1)$.

LEMMA 4.1. *If a resolvable $\text{OA}(n, m, s, 2)$ can be partitioned into r $\text{OA}(n/r, m, s, 1)$'s, then $m \leq (n - r)/(s - 1)$.*

Lemma 4.1 says the maximal value of m in a $\text{ROA}(n, m, s, 2; \alpha)$ is $[n - n/(\alpha s)]/(s - 1) = n[1 - 1/(\alpha s)]/(s - 1)$. If in Proposition 4.5 we let $\mathbf{A} = (0, \dots, s - 1)^T$ and $\mathbf{B}_i = \text{ROA}(n_2, m_2, s, 2; \alpha)$ with m_2 being the upper bound in Lemma 4.1 for a prime power s , namely, $m_2 = n_2[1 - 1/(\alpha s)]/(s - 1)$, we obtain Proposition 4.6, which states that if \mathbf{B}_i 's are α -resolvable orthogonal arrays of n_2 runs and the number of columns in each \mathbf{B}_i reaches the upper bound, α -resolvable orthogonal arrays of $n_2 s^k$ runs with the number of columns reaching the corresponding upper bound can be constructed, $k \geq 1$.

PROPOSITION 4.6. *If there exists a $\text{ROA}(n_2, m_2, s, 2; \alpha)$ with $m_2 = n_2[1 - 1/(\alpha s)]/(s - 1)$ for a prime power s , then a $\text{ROA}(n_2 s^k, m, s, 2; \alpha)$ with $m = n_2 s^k[1 - 1/(\alpha s)]/(s - 1)$ and $k \geq 1$ can be constructed.*

Applying Proposition 4.6 with $n_2 = s, m_2 = 1, \alpha = 1$, the proposed construction in (6) provides a $\text{CROA}(s^2, s, s, 2)$, which can be used as \mathbf{B}_i to obtain a $\text{CROA}(s^3, s^2, s, 2)$, and so on. By such repeated applications, a $\text{CROA}(s^k, s^{k-1}, s, 2)$ can be constructed via the construction in (6). The existing method, namely *difference schemes method*, can also construct a $\text{CROA}(s^k, s^{k-1}, s, 2)$ [21]. The advantage of the construction in (6) over the existing method is that it produces a large pool of such completely resolvable orthogonal arrays, which possibly allow designs with better properties to be found.

4.4. *Balanced sliced orthogonal arrays.* This section shows that using a sliced orthogonal array for \mathbf{A} and an orthogonal array for \mathbf{B}_i , the proposed construction in (6) provides a larger balanced sliced orthogonal array of the same levels and number of slices. Proposition 4.7 provides the detailed result. Without loss of generality, the levels of orthogonal arrays in this section and next section are all labeled by the elements in $\text{GF}(s)$ and δ is a level-collapsing projection on $\text{GF}(s)$ and satisfies $\delta(a + b) = \delta(a) + \delta(b)$. The truncation projection and modulus projection defined in [30], and subgroup projection defined in [38] are all level-collapsing projections on $\text{GF}(s)$ and satisfy $\delta(a + b) = \delta(a) + \delta(b)$.

PROPOSITION 4.7. *If $\mathbf{A} = (\mathbf{A}_1^T, \dots, \mathbf{A}_v^T)^T$ is an $\text{SOA}(n_1, m_1, s, 2; v, s_0)$ where each of $\mathbf{A}_1, \dots, \mathbf{A}_v$ is an $\text{OA}(n_0, m_1, s_0, 2)$ after some level-collapsing projection δ satisfying $\delta(a + b) = \delta(a) + \delta(b)$ for any two elements $a, b \in \text{GF}(s)$, and for $i = 1, \dots, n_1$, $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 1)$ for $m_2 = 1$ and $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 2)$ for $m_2 \geq 2$, then design \mathbf{E} in (6) is a $\text{BSOA}(n_1 n_2, (s - 1)m_1 m_2 + m_1 + m_2, s, 2; v, s_0)$ that can be expressed as $\mathbf{E} = (\mathbf{E}_1^T, \dots, \mathbf{E}_v^T)^T$ where each of $\mathbf{E}_1, \dots, \mathbf{E}_v$ becomes an $\text{OA}(n_0 n_2, (s - 1)m_1 m_2 + m_1 + m_2, s_0, 2)$ after the level-collapsing projection δ .*

TABLE 4
BSOA(16, 3, 4, 2; 4, 2)

		A			$\delta(\mathbf{A})$	
A ₁	0	0	0	0	0	0
	1	x	$x + 1$	1	0	1
	x	$x + 1$	1	0	1	1
	$x + 1$	1	x	1	1	0
A ₂	1	1	1	1	1	1
	0	$x + 1$	x	0	1	0
	$x + 1$	x	0	1	0	0
	x	0	$x + 1$	0	0	1
A ₃	x	x	x	0	0	0
	$x + 1$	0	1	1	0	1
	0	1	$x + 1$	0	1	1
	1	$x + 1$	0	1	1	0
A ₄	$x + 1$	$x + 1$	$x + 1$	1	1	1
	x	1	0	0	1	0
	1	0	x	1	0	0
	0	x	1	0	0	1

Proposition 4.7 can be shown using the same arguments as the proof of Theorem 3.1, and the fact that for any design \mathbf{D} , $\delta(\mathbf{D})$ is an $\text{OA}(n, m, s_0, 2)$ if \mathbf{D} is an $\text{OA}(n, m, s, 2)$, and under the condition $\delta(a + b) = \delta(a) + \delta(b)$, $\delta(\mathbf{D})$ is a $\text{D}(n, m, s_0)$ if \mathbf{D} is a $\text{D}(n, m, s)$. Examples 4 and 5 illustrate the use of the proposed construction for balanced sliced orthogonal arrays of run sizes being a prime power and not being a prime power, respectively.

EXAMPLE 4. Consider a $\text{BSOA}(16, 3, 4, 2; 4, 2)$ listed in Table 4. This array \mathbf{A} can be partitioned into $\mathbf{A}_1, \dots, \mathbf{A}_4$ each of which becomes an $\text{OA}(4, 3, 2, 2)$ after the level collapsing δ , where $\delta(0) = \delta(x) = 0, \delta(1) = \delta(x + 1) = 1$ and δ satisfies $\delta(a + b) = \delta(a) + \delta(b)$ for any two elements $a, b \in \text{GF}(4)$. Let $\mathbf{B}_i = \text{OA}(16, 5, 4, 2)$ for $i = 1, \dots, 16$, we obtain an $\mathbf{E} = \text{BSOA}(256, 53, 4, 2; 4, 2)$ that can be expressed as $\mathbf{E} = (\mathbf{E}_1^T, \dots, \mathbf{E}_4^T)^T$ with the property that each of $\mathbf{E}_1, \dots, \mathbf{E}_4$ becomes an $\text{OA}(64, 53, 2, 2)$ after the level-collapsing projection δ .

EXAMPLE 5. Consider an $\mathbf{A} = \text{BSOA}(81, 4, 9, 2; 9, 3)$ provided by [1]. This array \mathbf{A} can be partitioned into $\mathbf{A}_1, \dots, \mathbf{A}_9$ each of which becomes an $\text{OA}(9, 4, 3, 2)$ after the level collapsing δ , where $\delta(0) = \delta(x + 1) = \delta(2x + 2) = 0, \delta(1) = \delta(x + 2) = \delta(2x) = 1, \delta(2) = \delta(x) = \delta(2x + 1) = 2$ and δ satisfies $\delta(a + b) = \delta(a) + \delta(b)$ for any two elements $a, b \in \text{GF}(9)$. Let $\mathbf{B}_i = \text{OA}(162, 19, 9, 2)$ for $i = 1, \dots, 81$, we obtain an $\mathbf{E} = \text{BSOA}(13,122, 631, 9, 2; 9, 3)$ that can be expressed as $\mathbf{E} = (\mathbf{E}_1^T, \dots, \mathbf{E}_9^T)^T$ with the property that each of $\mathbf{E}_1, \dots, \mathbf{E}_9$ becomes an $\text{OA}(1458, 631, 3, 2)$ after the level-collapsing projection δ .

Table 5 lists the examples of balanced sliced orthogonal arrays constructed by the proposed method as in Proposition 4.7 and by [1] without our results. Comparing m_a and m in Table 5, the number of columns in balanced sliced orthogonal arrays constructed by [1] and the proposed method, it can be seen that the proposed construction results in more columns.

4.5. *Nested orthogonal arrays.* Proposition 4.8 shows that the proposed construction in (6) provides nested orthogonal arrays of larger run sizes using a nested orthogonal array for \mathbf{A} of n_1 runs and an orthogonal array for \mathbf{B}_i , for $i = 1, \dots, n_1$.

TABLE 5
Examples of balanced sliced orthogonal arrays constructed by [1] and the proposed method

n	s	s_0	v	m_a	m	n_1	m_1	n_2	m_2
64	4	2	4	12	13	16	3	4	1
256	4	2	4	48	53	16	3	16	5
512	4	2	8	48	53	32	3	16	5
1024	4	2	4	192	213	256	53	4	1
512	8	2	8	56	57	64	7	8	1
4096	8	2	8	448	457	64	7	64	9
729	9	3	9	36	37	81	4	9	1
6561	9	3	3	243	253	81	3	81	10

Where m_a and m represent the number of columns obtained by [1] and the proposed method, and n_1, n_2, m_1, m_2 are as defined in Proposition 4.7.

PROPOSITION 4.8. *If $\mathbf{A} = (\mathbf{A}_1^T, \mathbf{A}_2^T)^T$ is an $\text{NOA}(n_1, m_1, s, 2; n_0, s_0)$ where the array after applying a level-collapsing projection δ to \mathbf{A}_1 is an $\text{OA}(n_0, m_1, s_0, 2)$, and δ satisfies $\delta(a + b) = \delta(a) + \delta(b)$ for any two elements $a, b \in \text{GF}(s)$, and for $i = 1, \dots, n_1$ and a prime power $s > 2$, $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 1)$ for $m_2 = 1$ and $\mathbf{B}_i = \text{OA}(n_2, m_2, s, 2)$ for $m_2 \geq 2$, then design \mathbf{E} in (6) is an $\text{NOA}(n_1 n_2, (s - 1)m_1 m_2 + m_1 + m_2, s, 2; n_0 n_2, s_0)$ that can be expressed as $\mathbf{E} = (\mathbf{E}_1^T, \mathbf{E}_2^T)^T$ where \mathbf{E}_1 becomes an $\text{OA}(n_0 n_2, (s - 1)m_1 m_2 + m_1 + m_2, s_0, 2)$ after the level-collapsing projection δ .*

The proof of Proposition 4.8 is similar to that of Theorem 3.1, and thus we omit it. Example 6 gives an illustration of the use of Proposition 4.8.

EXAMPLE 6. Let \mathbf{A} be the $\text{OA}(16, 3, 4, 2)$ listed in Table 4 and δ defined in Example 4. It is easy to check $\delta(\mathbf{A}_1)$ being an $\text{OA}(4, 3, 2, 2)$, thus $\mathbf{A} = (\mathbf{A}_1^T, (\mathbf{A}_2^T, \mathbf{A}_3^T, \mathbf{A}_4^T))^T$ is an $\text{NOA}(16, 3, 4, 2; 4, 2)$. For $i = 1, \dots, 4$, taking $\mathbf{B}_i = \text{OA}(16, 5, 4, 2)$ from the orthogonal array website [36], we obtain an $\text{NOA}(256, 53, 4, 2; 64, 2)$ based on Proposition 4.8, and the array has more columns than an $\text{NOA}(256, 48, 4, 2; 64, 2)$ provided by [30]. Another example is, using an $\text{NOA}(64, 5, 8, 2; 16, 4)$ and an $\text{OA}(64, 9, 8, 2)$, we have an $\text{NOA}(4096, 329, 8, 2; 1024, 4)$. However, [30] can only give an $\text{NOA}(4096, 320, 8, 2; 1024, 4)$ with an $\text{NOA}(64, 5, 8, 2; 16, 4)$ and a $\text{D}(64, 64, 8)$.

5. Conclusion and discussion. We introduce a versatile method for constructing different kinds of orthogonal arrays. Armed with the theoretical results in Sections 3 and 4, we demonstrate that the proposed construction provides countless series of orthogonal arrays of strength two and three. Some examples of such arrays are listed in Table 2 and Example 2. Examples of new sliced orthogonal arrays and nested orthogonal arrays are given in Table 5 and Example 6. It shall be emphasized that, although the proposed method constructs nearly the same orthogonal arrays of strength two and three as the existing approaches in terms of the run sizes and the number of factors, it provides new orthogonal arrays of strength two and three by applying the proposed structure using nonisomorphic \mathbf{A} 's, nonisomorphic \mathbf{B}_i 's or isomorphic \mathbf{B}_i 's up to column permutations and/or level relabeling of one or more columns. The same statement applies to resolvable orthogonal arrays. Particularly, for orthogonal arrays of large run sizes, the proposed method provides a new, flexible and efficient way to find good designs. For complete results on new orthogonal arrays of strength two and three, resolvable orthogonal arrays, sliced orthogonal arrays and nested orthogonal arrays offered by the proposed construction, it deserves a comprehensive investigation, and thus we do not dwell on this issue here.

The proposed method is shown to construct balanced sliced orthogonal arrays, nested orthogonal arrays, and orthogonal arrays of (near) strength three, the last of which is a new class of orthogonal arrays we introduce. Compared with the existing approaches for constructing balanced sliced orthogonal arrays and nested orthogonal arrays, the proposed method provides new such arrays of run sizes that are not prime powers, noting that [1] only constructs the balanced sliced orthogonal arrays of run sizes being prime powers, and [31] and [38] only offer the nested orthogonal arrays of run sizes being prime powers.

One direction for future work is to apply the proposed approach to obtain optimal orthogonal arrays in a similar fashion as in [45], which combines the construction with the algorithmic procedure. Another direction is to use this structure to obtain the catalogue of optimal designs [26]. One direction that deserves exploration is the use of the proposed approach for constructing mixed-level orthogonal arrays, and its use in constructing strong orthogonal arrays [17, 51].

APPENDIX: PROOFS

PROOF OF LEMMA 3.1. It is easy to obtain items (i) and (iii). Under item (ii), by row permutations, $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \alpha \mathbf{b})$ can become n_2/s repetitions of $(\xi \oplus \mathbf{a}, \alpha \xi \oplus \mathbf{c})$, where $\xi = (\alpha_0, \alpha_1, \dots, \alpha_{s-1})^T$. Then items (ii)(1) to (ii)(3) follow from this structure. Under item (ii)(4), note that $(\xi, \alpha \xi)$ is a $D(s, 2, s)$ when $\alpha \neq 1$. In addition, as \mathbf{a} is balanced, then $(\xi \oplus \mathbf{a}, \alpha \xi \oplus \mathbf{a})$ is an orthogonal array. Hence the conclusion is true. \square

PROOF OF LEMMA 3.2. Item (i) is obvious according to the definition of generalized Kronecker sum and orthogonal arrays of strength three. Item (iv) is directly followed by Lemma 3.1. We show items (ii) and (iii), respectively. For items (ii) and (iii)(1) and (iii)(2), we prove that for any level combination (α^*, β^*) in the rows of $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \mathbf{d})$, the corresponding rows in $\mathbf{e} \circledast \mathbf{f}$ form a balanced design. Under item (ii), since $(\mathbf{b}_i, \mathbf{d}_i) = OA(n_2, 2, s, 2)$, for any pair of (α^*, β^*) , there are n_2/s^2 rows in $(a_i + \mathbf{b}_i, c_i + \mathbf{d}_i)$ equal to (α^*, β^*) , and for the corresponding rows in $e_i + \mathbf{f}_i = e_i + \alpha \mathbf{b}_i$ take the value of $\alpha \alpha^* + (e_i - \alpha a_i)$. Because $(\mathbf{e}, \alpha \mathbf{a}) = D(n_1, 2, s)$, that implies $\{\alpha \alpha^* + (e_i - \alpha a_i)\}_{i=1}^{n_1}$ take each value of $\text{GF}(s)$ with the same frequency, and hence those rows in $\mathbf{e} \circledast \mathbf{f}$ are balanced. Under item (iii), as $(\mathbf{a}, \mathbf{c}) = OA(n_1, 2, s, 2)$, we have $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \beta \mathbf{b})$ is an $OA(n_1 n_2, 2, s, 2)$ according to item (ii)(2) of Lemma 3.1, and for the rows in $(a_i + \mathbf{b}_i, c_i + \beta \mathbf{b}_i)$ equal to (α^*, β^*) , we have $c_i - \beta a_i = \beta^* - \beta \alpha^*$. If $\beta = 0$, it is clear that there are n_1/s rows in $(\beta \mathbf{a}, \mathbf{c})$ satisfying the relationship $c_i - \beta a_i = \beta^* - \beta \alpha^*$. If $\beta \neq 0$, as $(\beta \mathbf{a}, \mathbf{c}) = OA(n_1, 2, s, 2)$, there are n_1/s rows in $(\beta \mathbf{a}, \mathbf{c})$ satisfying the relationship $c_i - \beta a_i = \beta^* - \beta \alpha^*$, and both \mathbf{a} and \mathbf{c} are still balanced in such n_1/s rows. Correspondingly, in $\mathbf{e} \circledast \mathbf{f}$, we have $e_i + \alpha \mathbf{b}_i = e_i + \alpha(\alpha^* - a_i) = \alpha \alpha^* + (e_i - \alpha a_i)$ as $b_i = \alpha^* - a_i$. Under item (iii)(1), $e_i - \alpha a_i = (1 - \alpha)a_i$; or under item (iii)(2), $e_i - \alpha a_i = -\alpha a_i$, both are balanced in the corresponding n_1/s rows. Therefore, either in item (iii)(1) or (iii)(2), the corresponding rows in $\mathbf{e} \circledast \mathbf{f}$ are balanced. Thus, items (iii)(1) and (iii)(2) follow. Under item (iii)(3), consider $(\mathbf{a} \circledast \mathbf{b}, \mathbf{c} \circledast \beta \mathbf{b}, \mathbf{e} \circledast \alpha \mathbf{b})$. By row permutations, the array can be arranged as n_2/s repetitions of $(\xi \oplus \mathbf{a}, \beta \xi \oplus \mathbf{c}, \alpha \xi \oplus \mathbf{e})$, which is an $OA(sn_1, 3, s, 3)$ by $(\mathbf{a}, \mathbf{c}, \mathbf{e}) = OA(n_1, 3, s, 3)$, where $\xi = (\alpha_0, \alpha_1, \dots, \alpha_{s-1})^T$. Thus, item (iii)(3) follows. We complete the proof. \square

PROOF OF THEOREM 3.1. Theorem 3.1 can be verified by checking the orthogonality of any two columns in \mathbf{E} in (6) according to Lemma 3.1. More specifically, consider any two columns of \mathbf{E} , which can be represented by $\mathbf{u} = \mathbf{a}_p \circledast (\alpha_j * \mathbf{b}_q)$ and $\mathbf{v} = \mathbf{a}_{p'} \circledast (\alpha_{j'} * \mathbf{b}_{q'})$, Table 6 lists conditions in Lemma 3.1 used for all combinations of \mathbf{u} and \mathbf{v} . Note that $g = 1, \dots, s-1$ in the table. \square

TABLE 6
Conditions in Lemma 3.1 used for different \mathbf{u} and \mathbf{v}

\mathbf{u} and \mathbf{v}	(p, q) and (p', q')	Condition
$\mathbf{u} \in D_g$ and $\mathbf{v} \in D_g$	$q \neq q'$	(i)
	$p \neq p', q = q'$	(ii)(2)
$\mathbf{u} \in D_s, \mathbf{v} \in D_s$	$q \neq q'$	(i)
$\mathbf{u} \in \mathbf{A} \oplus \mathbf{0}_{n_2}, \mathbf{v} \in \mathbf{A} \oplus \mathbf{0}_{n_2}$	$p \neq p'$	(iii)(2)
$\mathbf{u} \in D_g$ and $\mathbf{v} \in D_{g'}, g \neq g'$	$q \neq q'$	(i)
	$p \neq p', q = q'$	(ii)(2)
	$p = p', q = q'$	(ii)(4)
$\mathbf{u} \in D_g$ and $\mathbf{v} \in D_s$	$q \neq q'$	(i)
	$q = q'$	(ii)(3)
$\mathbf{u} \in D_g, \mathbf{v} \in \mathbf{A} \oplus \mathbf{0}_{n_2}$		(iii)(1)
$\mathbf{u} \in D_s, \mathbf{v} \in \mathbf{A} \oplus \mathbf{0}_{n_2}$		(iii)(1)

Acknowledgments. We extend our sincere thanks to the Editor, Associate Editor and several referees for their constructive comments that lead to numerous improvements over the previous versions.

Funding. He was supported by National Natural Science Foundation of China, Grant No. 11701033.

Lin was supported by Discovery grant from the Natural Sciences and Engineering Research Council of Canada.

Sun is supported by Supported by National Natural Science Foundation of China Grant No. 11971098 and National Key Research and Development Program of China (No. 2020YFA0714102).

SUPPLEMENTARY MATERIAL

Supplement to “A new and flexible design construction for orthogonal arrays for modern applications” (DOI: [10.1214/21-AOS2159SUPP](https://doi.org/10.1214/21-AOS2159SUPP); .pdf). We provide results on determining the maximal number of columns in \mathbf{E} in (6) for given values of n, s and t , and new results on lower bounds of the maximal number of columns in certain orthogonal arrays of strength three.

REFERENCES

- [1] AI, M., JIANG, B. and LI, K. (2014). Construction of sliced space-filling designs based on balanced sliced orthogonal arrays. *Statist. Sinica* **24** 1685–1702. [MR3308657](#)
- [2] BOSE, R. C. and BUSH, K. A. (1952). Orthogonal arrays of strength two and three. *Ann. Math. Stat.* **23** 508–524. [MR0051204](#) <https://doi.org/10.1214/aoms/1177729331>
- [3] DENG, X., HUNG, Y. and LIN, C. D. (2015). Design for computer experiments with qualitative and quantitative factors. *Statist. Sinica* **25** 1567–1581. [MR3409081](#)
- [4] DENG, X., LIN, C. D., LIU, K. W. and ROWE, R. K. (2017). Additive Gaussian process for computer models with qualitative and quantitative factors. *Technometrics* **25** 1567–1581.
- [5] DEY, A. and MUKERJEE, R. (1999). *Fractional Factorial Plans*. Wiley Series in Probability and Statistics: Probability and Statistics. Wiley, New York. [MR1679441](#) <https://doi.org/10.1002/9780470316986>
- [6] EENDEBAK, P. and SCHOEN, E. (2017). Complete series of non-isomorphic orthogonal arrays. Available at <http://www.pietereendebak.nl/oapackage/series.html>.
- [7] GOH, J., BINGHAM, D., HOLLOWAY, J. P., GROSSKOPF, M. J., KURANZ, C. C. and RUTTER, E. (2013). Prediction and computer model calibration using outputs from multifidelity simulators. *Technometrics* **55** 501–512. [MR3176554](#) <https://doi.org/10.1080/00401706.2013.838910>
- [8] GOPALAKRISHNAN, K. and STINSON, D. R. (2006). Applications of orthogonal arrays to computer science. In *Proceeding of ICDM* 149–164.

- [9] GUPTA, V. K., NIGAM, A. K. and DEY, A. (1982). Orthogonal main-effect plans for asymmetrical factorials. *Technometrics* **24** 135–137. [MR0655577](#) <https://doi.org/10.2307/1268491>
- [10] HAALAND, B. and QIAN, P. Z. G. (2011). Accurate emulators for large-scale computer experiments. *Ann. Statist.* **39** 2974–3002. [MR3012398](#) <https://doi.org/10.1214/11-AOS929>
- [11] HE, X. and QIAN, P. Z. G. (2014). A central limit theorem for general orthogonal array based space-filling designs. *Ann. Statist.* **42** 1725–1750. [MR3262466](#) <https://doi.org/10.1214/14-AOS1231>
- [12] HE, X. and QIAN, P. Z. G. (2016). A central limit theorem for nested or sliced Latin hypercube designs. *Statist. Sinica* **26** 1117–1128. [MR3559945](#)
- [13] HE, Y., LIN, C. D. and SUN, F. (2017). On construction of marginally coupled designs. *Statist. Sinica* **27** 665–683. [MR3674691](#)
- [14] HE, Y., LIN, C. D. and SUN, F. (2019). Construction of marginally coupled designs by subspace theory. *Bernoulli* **25** 2163–2182. [MR3961244](#) <https://doi.org/10.3150/18-BEJ1049>
- [15] HE, Y., LIN, C. D. and SUN, F. (2022). Supplement to “A new and flexible design construction for orthogonal arrays for modern applications.” <https://doi.org/10.1214/21-AOS2159SUPP>
- [16] HE, Y., LIN, C. D., SUN, F. and LV, B. (2017). Marginally coupled designs for two-level qualitative factors. *J. Statist. Plann. Inference* **187** 103–108. [MR3638045](#) <https://doi.org/10.1016/j.jspi.2017.02.010>
- [17] HE, Y. and TANG, B. (2013). Strong orthogonal arrays and associated Latin hypercubes for computer experiments. *Biometrika* **100** 254–260. [MR3034340](#) <https://doi.org/10.1093/biomet/ass065>
- [18] HEDAYAT, A. S. (1986). Orthogonal arrays of strength $t+$ and their statistical applications. Technical report, Univ. Illinois, Chicago, IL.
- [19] HEDAYAT, A. S. (1990). New properties of orthogonal arrays and their statistical applications. In *Statistical Design and Analysis of Industrial Experiments* (S. Ghosh, ed.) 407–422. Dekker, New York.
- [20] HEDAYAT, A. S. (2017). A scientific tour on orthogonal arrays. In *Mathematics Across Contemporary Sciences* (T. Abualrub, A. Jarrah, S. Kallel and H. Sulieman, eds.). *Springer Proc. Math. Stat.* **190** 111–135. Springer, Cham. [MR3630785](#) https://doi.org/10.1007/978-3-319-46310-0_7
- [21] HEDAYAT, A. S., SLOANE, N. J. A. and STUFKEN, J. (1999). *Orthogonal Arrays: Theory and Applications*. *Springer Series in Statistics*. Springer, New York. [MR1693498](#) <https://doi.org/10.1007/978-1-4612-1478-6>
- [22] HUANG, H., LIN, D. K. J., LIU, M.-Q. and YANG, J.-F. (2016). Computer experiments with both qualitative and quantitative variables. *Technometrics* **58** 495–507. [MR3556617](#) <https://doi.org/10.1080/00401706.2015.1094416>
- [23] HWANG, Y., QIAN, P. Z. G. and HE, X. (2016). Sliced orthogonal array-based Latin hypercube designs. *Technometrics* **58** 50–61. [MR3463156](#) <https://doi.org/10.1080/00401706.2014.993092>
- [24] KONG, X., AI, M. and TSUI, K. L. (2018). Flexible sliced designs for computer experiments. *Ann. Inst. Statist. Math.* **70** 631–646. [MR3785710](#) <https://doi.org/10.1007/s10463-017-0603-3>
- [25] LEKIVETZ, R., SITTER, R., BINGHAM, D., HAMADA, M. S., MOORE, L. M. and WENDELBERGER, J. R. (2015). On algorithms for obtaining orthogonal and near-orthogonal arrays for main-effects screening. *J. Qual. Technol.* **47** 2–13.
- [26] LIN, C. D., SITTER, R. R. and TANG, B. (2012). Creating catalogues of two-level nonregular fractional factorial designs based on the criteria of generalized aberration. *J. Statist. Plann. Inference* **142** 445–456. [MR2843048](#) <https://doi.org/10.1016/j.jspi.2011.08.003>
- [27] MUKERJEE, R., SUN, F. and TANG, B. (2014). Nearly orthogonal arrays mappable into fully orthogonal arrays. *Biometrika* **101** 957–963. [MR3286928](#) <https://doi.org/10.1093/biomet/asu042>
- [28] MUKERJEE, R. and WU, C. F. J. (2006). *A Modern Theory of Factorial Designs*. *Springer Series in Statistics*. Springer, New York. [MR2230487](#)
- [29] OWEN, A. B. (1992). Orthogonal arrays for computer experiments, integration and visualization. *Statist. Sinica* **2** 439–452. [MR1187952](#)
- [30] QIAN, P. Z. G., AI, M. and WU, C. F. J. (2009). Construction of nested space-filling designs. *Ann. Statist.* **37** 3616–3643. [MR2549572](#) <https://doi.org/10.1214/09-AOS690>
- [31] QIAN, P. Z. G., TANG, B. and WU, C. F. J. (2009). Nested space-filling designs for computer experiments with two levels of accuracy. *Statist. Sinica* **19** 287–300. [MR2487890](#)
- [32] QIAN, P. Z. G. and WU, C. F. J. (2009). Sliced space-filling designs. *Biometrika* **96** 945–956. [MR2767280](#) <https://doi.org/10.1093/biomet/asp044>
- [33] QIAN, P. Z. G., WU, H. and WU, C. F. J. (2008). Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics* **50** 383–396. [MR2457574](#) <https://doi.org/10.1198/004017008000000262>
- [34] SACKS, J., WELCH, W. J., MITCHELL, T. J. and WYNN, H. P. (1989). Design and analysis of computer experiments. *Statist. Sci.* **4** 409–435. [MR1041765](#)
- [35] SANTNER, T. J., WILLIAMS, B. J. and NOTZ, W. I. (2018). *The Design and Analysis of Computer Experiments*. *Springer Series in Statistics*. Springer, New York. [MR3887662](#)

- [36] SLOANE, N. J. A. (2004). A library of orthogonal arrays. Available at <http://neilsloane.com/oadir/>.
- [37] SUEN, C.-Y. (1989). Some resolvable orthogonal arrays with two symbols. *Comm. Statist. Theory Methods* **18** 3875–3881. MR1040681 <https://doi.org/10.1080/03610928908830128>
- [38] SUN, F., LIU, M.-Q. and QIAN, P. Z. G. (2014). On the construction of nested space-filling designs. *Ann. Statist.* **42** 162–193. MR3226161 <https://doi.org/10.1214/14-AOS1229>
- [39] SUN, F., WANG, Y. and XU, H. (2019). Uniform projection designs. *Ann. Statist.* **47** 641–661. MR3909945 <https://doi.org/10.1214/18-AOS1705>
- [40] TANG, B. (1993). Orthogonal array-based Latin hypercubes. *J. Amer. Statist. Assoc.* **88** 1392–1397. MR1245375
- [41] WANG, J. C. and WU, C. F. J. (1992). Nearly orthogonal arrays with mixed levels and small runs. *Technometrics* **34** 409–422. MRMR1939683
- [42] WEI, Y. and XIONG, S. (2019). Bayesian integrative analysis for multi-fidelity computer experiments. *J. Appl. Stat.* **46** 1973–1987. MR3963046 <https://doi.org/10.1080/02664763.2019.1575340>
- [43] WU, C. F. J. and HAMADA, M. S. (2009). *Experiments: Planning, Analysis, and Optimization*, 2nd ed. *Wiley Series in Probability and Statistics*. Wiley, Hoboken, NJ. MR2583259
- [44] XU, H. (2002). An algorithm for constructing orthogonal and nearly-orthogonal arrays with mixed levels and small runs. *Technometrics* **44** 356–368. MR1939683 <https://doi.org/10.1198/004017002188618554>
- [45] XU, H., CHENG, S.-W. and WU, C. F. J. (2004). Optimal projective three-level designs for factor screening and interaction detection. *Technometrics* **46** 280–292. MR2082498 <https://doi.org/10.1198/004017004000000310>
- [46] XU, H., PHOA, F. K. H. and WONG, W. K. (2009). Recent developments in nonregular fractional factorial designs. *Stat. Surv.* **3** 18–46. MR2520978 <https://doi.org/10.1214/08-SS040>
- [47] YANG, F., LIN, C. D., ZHOU, Y. D. and HE, Y. (2021). Doubly coupled designs for computer experiments with both qualitative and quantitative factors. *Statist. Sinica.* <https://doi.org/0.5705/ss.202020.0317>
- [48] YANG, J.-F., LIN, C. D., QIAN, P. Z. G. and LIN, D. K. J. (2013). Construction of sliced orthogonal Latin hypercube designs. *Statist. Sinica* **23** 1117–1130. MR3114707
- [49] YANG, X., CHEN, H. and LIU, M.-Q. (2014). Resolvable orthogonal array-based uniform sliced Latin hypercube designs. *Statist. Probab. Lett.* **93** 108–115. MR3244562 <https://doi.org/10.1016/j.spl.2014.06.021>
- [50] ZHANG, Y., TAO, S., CHEN, W. and APLEY, D. W. (2020). A latent variable approach to Gaussian process modeling with qualitative and quantitative factors. *Technometrics* **62** 291–302. MR4125497 <https://doi.org/10.1080/00401706.2019.1638834>
- [51] ZHOU, Y. and TANG, B. (2019). Column-orthogonal strong orthogonal arrays of strength two plus and three minus. *Biometrika* **106** 997–1004. MR4031213 <https://doi.org/10.1093/biomet/asz043>