Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Flexible Design Strategies and Position Modeling for Order-of-Addition Screening Experiments

Bing Wen
Chinese Academy of Sciences
Bo Hu
hub4207@dufe.edu.cn
Dongbei University of Finance and Economics

Research Article

Keywords: component-position screening model, D-efficiency, order-of-addition screening experiment, sequence selection

Posted Date: August 28th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10523232/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Flexible Design Strategies and Position Modeling for Order-of-Addition Screening Experiments

Bing Wen[1] and Bo Hu[2*]

[1]Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Zhongguancun East Road, No. 55, Beijing, 100190, China.
[2*]School of Statistics, Dongbei University of Finance and Economics, Jianshan Street, No. 217, Dalian, 116025, Liaoning, China.

*Corresponding author(s). E-mail(s): hub4207@dufe.edu.cn;
Contributing authors: wenbing@amss.ac.cn;

Abstract

Order-of-addition screening experiments arise when only a subset of the available components can be accommodated in each run, so that component selection and sequence must be determined simultaneously. Existing screening models either require many parameters or do not directly represent nonlinear absolute-position effects. We introduce linear, quadratic, and second-order component-position screening models (LCPS, QCPS, and SCPS) that use position zero to encode omission and allow the model complexity to match the experimental budget. We establish D-, G-, and I-optimality of the full screening design for LCPS and QCPS, show that several existing constructions remain optimal under the proposed models, and provide new constructions for general component and run-size settings. The proposed framework permits nonlinear position modeling in budget regimes where standard component-position and pairwise-ordering screening models are not estimable without regularisation. Numerical experiments evaluate not only D-efficiency but also prediction, active-effect recovery, and the identification of near-optimal screened sequences.

Keywords: component-position screening model, D-efficiency, order-of-addition screening experiment, sequence selection

1 Introduction

Order-of-addition experiments arise in chemical synthesis, food science, pharmaceutical research, and many operational systems. In these experiments, the response depends on the sequence in which materials, hereafter called *components*, are added. With m components, a complete experiment contains $m!$ sequences and is quickly infeasible. The central questions are therefore which sequences to run and how to model the resulting response under realistic budget constraints.

Existing literature on subset selection and the modeling of sequence effects has primarily been divided into two categories. One method is modeling based on the relative-position of components, called pairwise ordering (PWO) model (Van Nostrand, 1995). The theoretical results about optimal criteria and optimal designs under PWO model can be found in Chen et al. (2020); Peng et al. (2019); Schoen and Mee (2023); Tsai (2022); Voelkel (2019). Another method for analyzing order-of-addition experiments is to model based on the absolute-position of components, typically component-position (CP) model (Yang et al., 2021) and absolute-position effect models (Stokes and Xu, 2022). The design methods and optimization theories under these types of models can refer to Huang (2021); Wang et al. (2024); Zhao et al. (2021a,b), and so on.

In real-world applications, it is fairly common for the number of components m to exceed the number of available positions, q, in the sequence. Therefore, it is necessary to identify which components are important, and then determine the order in which they should be added. Such experiments are referred to as order-of-addition screening experiments. Stokes and Xu (2024) presented a framework for constructing optimal and near-optimal screening designs under the pairwise ordering screening (PWOS) model and the component-position screening (CPS) model, respectively. The PWOS model is given by

$$y = \beta_0 + \sum_{i<j} \beta_{ij} I_{ij} + \varepsilon,$$

where β_0 is the intercept, each factor I_{ij} takes levels 1 and -1 indicating whether or not component i is added before component j, and it takes value 0 if i or j is missing from the sequence, β_{ij} is the corresponding effect, random error $\varepsilon \sim N(0, \sigma^2)$. The CPS model is expressed as

$$y = \gamma_0 + \sum_{k=1}^{m-1} \sum_{j=1}^{q} z_{kj} \gamma_{kj} + \epsilon,$$

where γ_0 is the intercept, z_{kj} is an indicator for the component-position pair (k, j), γ_{kj} is the effect parameter of component k being added at position j. However, the PWOS model is unable to capture second-order sequence effects, while CPS model involves a large number of parameters. Additionally, designs under these methods have limitations of the number of runs, components and available positions.

To address these limitations, we propose flexible absolute-position screening models with a number of estimable parameters that can be matched to the available budget. The models accommodate linear, quadratic, and second-order position effects while

treating an omitted component as having position zero. We demonstrate that designs obtained by Stokes and Xu (2024) are D-, G-, and I-optimal under the proposed models and provide additional constructions for more general component and run-size settings. Because information-matrix efficiency is only a surrogate for the scientific goal of a screening experiment, our numerical study also evaluates out-of-sample prediction, recovery of active effects, and the regret incurred when a fitted model is used to select a sequence.

The remainder of the paper is organized as follows. Section 2 presents the novel absolute-position effects screening models and establishes the optimality of the full design. Section 3 shows that several existing designs are optimal under the proposed models and gives new constructions. Section 4 presents efficient screening designs and some numerical studies showed in Section 5. Section 6 concludes the paper. Proofs and supplementary D-efficiency results are provided in the Appendix.

2 Flexible component-position screening models and full design optimality

In this study, we assume that the number of available positions in the experiment is fixed at q, $1 < q < m$, with the m components denoted as $1, 2, \ldots, m$. Under this experimental setup, there are total $\binom{m}{q}q! = m!/(m-q)!$ possible ways to assign the m components to q positions. The design including every possible sequence is full screening design, denoted by $S_{m,q}$. Each design for order-of-addition experiment is specified by a component matrix $A = (a_{ij})_{n\times q}$, in which each element a_{ij} represents the component added in position j of the ith run. Its corresponding position matrix is defined as $B = (b_{ik})_{n\times m}$, where $b_{ik} = j$ if $a_{ij} = k$ for $k = 1, 2, \ldots, m$, $j = 1, 2, \ldots, q$, and $b_{ik} = 0$ if k do not appear in the ith run of A. For example, for a row in A $(m = 5, q = 4)$, $(1, 5, 2, 3)$, its corresponding row in B is then $(1, 3, 4, 0, 2)$.

2.1 Flexible component-position screening models

Effective analysis of order-of-addition screening experiments requires a suitable model that supports both design and analysis objectives. In practice, it is often observed that, in addition to main effects, a few two-factor interactions may also be significant. To address such complexities, Stokes and Xu (2022) proposed a class of position-based models and corresponding designs that are capable of handling increasingly complex scenarios–such as the five-drug example – without requiring an excessive number of runs under the classic *order-of-addition experiments*. Therefore, incorporating position-based modeling into the analysis of *order-of-addition screening experiments* is both necessary and beneficial.

The position-based model, in its original iteration, is not directly applicable to order-of-addition screening experiments. Therefore, we slightly modify it to suit for the order-of-addition screening experiments. Specifically, we define the linear component-position screening (LCPS) model (1), quadratic component-position screening (QCPS) model (2), and second-order component-position screening (SCPS) model (3) as

follows,

$$y = \mu_0 + \sum_{k=1}^{m-1} \mu_k b_k + \epsilon, \tag{1}$$

$$y = \mu_0 + \sum_{k=1}^{m-1} \mu_k b_k + \sum_{k=1}^{m-1} \mu_{kk} b_k^2 + \epsilon, \tag{2}$$

$$y = \mu_0 + \sum_{k=1}^{m-1} \mu_k b_k + \sum_{k=1}^{m-2} \mu_{kk} b_k^2 + \sum_{1 \le k < k' \le m-1} \mu_{kk'} b_k b_{k'} + \epsilon, \tag{3}$$

where y is the response, μ_0 is the intercept term, b_k denotes the position of component k, taking values from 0 to q, with 0 indicating that component k does not appear, the term μ_k represents the linear effect with the position b_k, μ_{kk} denotes the effect associated with the quadratic term b_k^2, $b_k b_{k'}$ represents the second-order interaction term, $\mu_{kk'}$ denotes its corresponding effect, and the error term $\epsilon \sim N(0, \sigma^2)$.

The matrix form of the models above can be written as

$$Y = \mathbf{1}_n \mu_0 + B_l \boldsymbol{\mu}_l + \boldsymbol{\epsilon}, \tag{4}$$

$$Y = \mathbf{1}_n \mu_0 + B_l \boldsymbol{\mu}_l + B_q \boldsymbol{\mu}_q + \boldsymbol{\epsilon}, \tag{5}$$

$$Y = \mathbf{1}_n \mu_0 + B_l \boldsymbol{\mu}_l + B_q \boldsymbol{\mu}_q + B_s \boldsymbol{\mu}_s + \boldsymbol{\epsilon}, \tag{6}$$

where Y is the n-dimensional response vector, $\mathbf{1}_n$ is an n-dimensional vector of ones, $B_l = (\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{m-1})$ denotes the matrix of all linear effect terms with $\boldsymbol{b}_j = (b_{1j}, \ldots, b_{nj})^T$, $\boldsymbol{\mu}_l = (\mu_1, \mu_2, \ldots, \mu_{m-1})^T$ is the associated coefficient vector, $B_q = (\boldsymbol{b}_1^2, \boldsymbol{b}_2^2, \ldots, \boldsymbol{b}_{m-1}^2)$ denotes the matrix of all quadratic effect terms with $\boldsymbol{b}_k^2 = (b_{1k}^2, b_{2k}^2, \ldots, b_{nk}^2)^T$ for $k = 1, 2, \ldots, m-1$, the vector $\boldsymbol{\mu}_q = (\mu_{11}, \mu_{22}, \ldots, \mu_{(m-1)(m-1)})^T$ represents the corresponding quadratic coefficient, $B_s = (\boldsymbol{b}_1\boldsymbol{b}_2, \boldsymbol{b}_1\boldsymbol{b}_3, \ldots, \boldsymbol{b}_{m-2}\boldsymbol{b}_{m-1})$ denotes the matrix of second-order interaction terms, with $\boldsymbol{b}_k\boldsymbol{b}_{k'} = (b_{1k}b_{1k'}, b_{2k}b_{2k'}, \ldots, b_{nk}b_{nk'})^T$ for $k < k'$, $k, k' = 1, 2, \ldots, m-1$, $\boldsymbol{\mu}_s = (\mu_{12}, \mu_{13}, \ldots, \mu_{(m-2)(m-1)})^T$ represents the interaction coefficients, and the error term is represented by the n-dimensional vector $\boldsymbol{\epsilon}$. Then the model matrix for LCPS model (1), QCPS model (2), SCPS model (3) can be expressed as $X_l = (\mathbf{1}_n, B_l)$, $X_q = (\mathbf{1}_n, B_l, B_q)$, $X_s = (\mathbf{1}_n, B_l, B_q, B_s)$, respectively.

In Table 1, we list the number of parameters in different models for various values of m and q, including LCPS, QCPS, SCPS, PWOS, and CPS model. The LCPS and QCPS models have fewer parameters than the others when $m > 4$. The SCPS model contains slightly more parameters than the PWOS model. Compared to the CPS model, SCPS model includes more parameters when $q \le \lfloor \frac{m+2}{2} - \frac{1}{m-1} \rfloor$, and fewer otherwise. The new position models are both parsimonious and flexible.

2.2 Full design optimality

Under the proposed models, designs are usually evaluated using optimal criteria. The most commonly used ones include D-, G-, and I-optimal criteria. For any n-run design D, let X denote the model matrix corresponding to the chosen model, and define the normalized information matrix as $M(D) = X^T X/n$. Let X_f denote the model

Table 1 The number of parameters in different models for various values of m and q.

Models		LCPS	QCPS	SCPS	CPS	PWOS
Parameter settings		m	$2m-1$	$\frac{(m-1)(m+2)}{2}$	$(m-1)q+1$	$\frac{m(m-1)}{2}+1$
$m=3$	$q=2$	3	5	5	5	4
$m=4$	$q=2$	4	7	9	7	7
	$q=3$	4	7	9	10	7
$m=5$	$q=2$	5	9	14	9	11
	$q=3$	5	9	14	13	11
	$q=4$	5	9	14	17	11
$m=6$	$q=2$	6	11	20	11	16
	$q=3$	6	11	20	16	16
	$q=4$	6	11	20	21	16
	$q=5$	6	11	20	26	16
$m=7$	$q=2$	7	13	27	13	22
	$q=3$	7	13	27	19	22
	$q=4$	7	13	27	25	22
	$q=5$	7	13	27	31	22
	$q=6$	7	13	27	37	22

matrix of full design, and $x \in X_f$ represents a row of X_f. Then (i) the D-optimal design maximizes $|M|$; (ii) the G-optimal design minimizes $\max_{x \in X_f} xM^{-1}x^T$; (iii) the I-optimal design minimizes $tr(X_f M^{-1} X_f^T)$. Our goal is to construct parsimonious and efficient order-of-addition screening designs. To this end, it is necessary to select a benchmark, i.e., the full design, against which other designs can be compared. The following results are established to facilitate this selection.

Theorem 1 *The full design $S_{m,q}$ is D-, G-, and I-optimal for LCPS model (1), $m \geq 3$, $1 < q < m$.*

Theorem 2 *The full design $S_{m,q}$ is D-, G-, and I-optimal for QCPS model (2), $m \geq 3$, $1 < q < m$.*

These results are not surprising, as full designs are often theoretically optimal. However, they may be impractical in many real-world scenarios due to resource constraints. To identify designs that are both practical and efficient, we turn to the concept of efficiency. For any design ξ, its D-efficiency under the chosen model, relative to the full design $S_{m,q}$ is defined as

$$D_{\text{eff}}(\xi) = \{|M(\xi)|/|M(S_{m,q})|\}^{1/p},$$

where p denotes the number of model parameters. Specifically, we refer to the D-efficiency of a design under the LCPS model (1), QCPS model (2), and SCPS model (3) as the linear D-efficiency ($D_{\text{l,eff}}$), quadratic D-efficiency ($D_{\text{q,eff}}$), and the second-order D-efficiency ($D_{\text{s,eff}}$), respectively. In the following sections, we introduce several methods for constructing efficient order-of-addition screening designs under these new models.

3 Optimal design construction

We first show that some existing designs are optimal under the proposed models, and then construct new and optimal order-of-addition screening designs under the new models.

3.1 Existing optimal designs

Before presenting the theoretical results, we first introduce some preliminaries, including key concepts and notation.

Definition 1 (Hedayat et al. (1999), P154) A cyclic Latin square is an $n \times n$ matrix in which each row is obtained by cyclically shifting every entry in the previous row one place to the right. A retrocirculant Latin square is an $n \times n$ matrix in which each row is obtained by cyclically shifting every entry in the previous row one place to the left. The design obtained with the first row $a = (a_1, \ldots, a_n)$ in these two ways are denoted as L_{a+} and L_{a-}, respectively. Let L_a denote either design L_{a+} or L_{a-}.

For a prime or prime power m, let $GF(m) = \{\omega_0, \omega_1, ..., \omega_{m-1}\}$ be a Galois field of order m with ω_0 being the zero element. When m is a prime, $GF(m) = \{0, 1, ..., m-1\}$ is a ring of integers modulo m.

Definition 2 (Yang et al. (2021)) For $n = \lambda m(m-1)$, an $n \times m$ matrix with entries from $\{1, 2, ..., m\}$ is called a component orthogonal array of n runs and m factors if each row is a permutation of $\{1, 2, ..., m\}$, and for any two columns, each of the level combinations (i, j) with $i \neq j$ and $i, j = 1, 2, \ldots, m$ appears λ times. We use $COA(n, m)$ to denote such a component orthogonal array and call the positive integer λ as the index.

For order-of-addition screening experiments, Stokes and Xu (2024) provided several algorithms to construct optimal designs under the CPS and PWOS models. We review these algorithms and summarize their optimality, which corresponds to Algorithm 1 and Theorem 3, as well as Algorithm 2 and Theorem 4, respectively. Their proofs could be found in Appendix.

Theorem 3 *For $n \leq m!$, $1 < q < m$, with n divisible by $m(m-1)$ and m being a prime or prime power, design $S^{CP}_{n,m,q}$ from Algorithm 1 is D-, G-, and I-optimal under the LCPS model (1).*

Remark 1 With the same parameter settings, design $S^{CP}_{n,m,q}$ above is also D-, G-, and I-optimal under the QCPS model (2).

Theorem 4 *For $n = 3\binom{m}{3}$, m even and $q = 3$, design $S^{PWO}_{n,m,3}$ from Algorithm 2 is D-, G-, and I-optimal under the LCPS model (1).*

Algorithm 1 Algorithm 1 in Stokes and Xu (2024), with m a prime power, $m(m-1)|n$.

1. Define matrix $L_k = (\omega_i + \omega_k * \omega_j)_{ij}$, $i, j = 0, \ldots, m-1$, $k = 1, \ldots, m-1$, $GF(m) = \{\omega_0, \ldots, \omega_{m-1}\}$.
2. Define $C_1 = (L_1^T, \ldots, L_{m-1}^T)^T$. Keep its first two columns fixed and permute the last $m-2$ columns, to obtain $(m-2)!$ permuted matrices, $C_1, \ldots, C_{(m-2)!}$.
3. Define $F_m = (C_1^T, \ldots, C_{m-1}^T)^T$ and replace ω_i with number i, $i = 0, \ldots, m-1$.
4. Let $F_{n,m}$ be the $n \times m$ design formed by the first n rows of F_m.
5. Construct an $n \times q$ matrix $S^{CP}_{n,m,q}$ by taking the first q odd-numbered columns of $F_{n,m}$ if $q \leq m/2$. Otherwise take the $\lceil m/2 \rceil$ odd-numbered columns followed by the first $q - \lceil m/2 \rceil$ even-numbered columns, where $\lceil x \rceil$ is the smallest integer that is equal to or larger than x.
6. Permute the columns of $S^{CP}_{n,m,q}$ to improve its performance under a chosen criterion.

Algorithm 2 Algorithm 2 in Stokes and Xu (2024), with m even, $n = 3\binom{m}{3}$.

1. Generate $\binom{m}{3}$ 3-component combinations (i, j, k) with $0 \leq i < j < k \leq m-1$.
2. For each 3-component combination such that $i+j+k$ is even, construct $D_{(i,k,j)+}$.
3. For each 3-component combination such that $i+j+k$ is odd, construct $D_{(i,j,k)-}$.
4. Construct $S^{PWO}_{n,m,3}$ by first row-wise concatenating the D_{ijk} for all $0 \leq i < j < k \leq m-1$ such that $i+j+k$ is even, then concatenating the remaining D_{ijk} and taking the first n rows.

These theoretical results establish the optimality of existing designs under the new proposed models, thereby indirectly demonstrating the practical relevance and applicability of the new modeling framework. Design $S^{CP}_{n,m,q}$ requires m to be a prime or prime power, while $S^{PWO}_{n,m,3}$ is limited to the case $q = 3$. These limitations highlight the need for new construction methods.

3.2 Optimal designs construction

Here, we present several methods for constructing optimal order-of-addition screening designs under the proposed models. First, for non-prime power m with $1 < q \leq m$, Algorithm 3 can be used to construct optimal designs under both LCPS model (1) and QCPS model (2). Algorithm 3 removes the restriction that m is a power of prime and the optimality is given in Theorem 5 and Remark 2. As their proof is similar to the previous one, it is omitted here for brevity.

Theorem 5 *For non-prime power m, $q \leq m$, design $S^{COA}_{n,m,q}$ from Algorithm 3 is D-, G-, and I-optimal under LCPS model (1), with run size $m!/([m]_p - 2)!$.*

Remark 2 With the same parameter settings, design $S^{COA}_{n,m,q}$ from Algorithm 3 is also D-, G-, and I-optimal under QCPS model (2).

Algorithm 3

1. Construct a design $D : COA(n = \lambda m(m-1), m)$ by recursively using (a)-(b) $m - [m]_p$ times with a $COA([m]_p([m]_p - 1), [m]_p)$ as the base design for the first iteration, where $[m]_p$ is the largest power of prime that does not exceed m and $\lambda = (m-2)!/([m]_p - 2)!$.
   (a) Suppose that $C = (c_1^T, \ldots, c_{n'}^T)^T$ is a $COA(\lambda' m'(m'-1), m')$. Denote $E = (e_1^T, \ldots, e_{n'}^T)^T$, where $e_i = (c_i, m')$ for $i = 1, \ldots, n'$.
   (b) For $i = 1, \ldots, n'$, construct $L_{e_i -}$, and define $G = (L_{e_1 -}^T, \ldots, L_{e_{n'} -}^T)^T$. Then G is a $COA(\lambda'(m'+1)m'(m'-1), m'+1)$.
2. Select q columns from D to optimize a chosen criterion. The resulting design is denoted by $S_{n,m,q}^{COA}$.

Moreover, we provide Algorithm 4 to generate optimal order-of-addition screening designs that are suitable for the case with $m \geq 3$, prime $2 \leq q < m$. In some cases, the run size is small and flexible. And their optimality results are discussed in Theorem 6. The following Example 1 is provided to illustrate Algorithm 4.

Algorithm 4

1. Generate all q-component combinations to from set $\mathcal{A}$=$\{\alpha : (l_0, \ldots, l_{q-1}), 1 \leq l_0 < \cdots < l_{q-1} \leq m\}$.
2. If $q = 2$, execute (a); otherwise, execute (b).
   (a) Construct the L_α for each $\alpha = (l_0, l_1)$ in $\mathcal{A}$, denoted as D_i, $i = 1, \ldots, \binom{m}{q}$.
   (b) Construct $\frac{q-1}{2}$ generating vectors $\alpha_1, \ldots, \alpha_{\frac{q-1}{2}}$ and its corresponding $L_{\alpha_1}, \ldots, L_{\alpha_{\frac{q-1}{2}}}$ for each $\alpha = (l_0, l_1, \ldots, l_{q-1}) \in \mathcal{A}$, where

$$\begin{aligned}
&\alpha_1 = (l_{0\dot{+}1}, l_{0\dot{+}2}, \ldots, l_{0\dot{+}q}),\\
&\alpha_2 = (l_{0\dot{+}2}, l_{0\dot{+}4}, \ldots, l_{0\dot{+}2q}),\\
&\ldots\\
&\alpha_{(q-1)/2} = (l_{0\dot{+}(q-1)/2}, l_{0\dot{+}2(q-1)/2}, \ldots, l_{0\dot{+}q(q-1)/2}),
\end{aligned}$$

   $x \dot{+} y = x + y \pmod q$. And define $D_i = \left(L_{\alpha_1}^T, \ldots, L_{\alpha_{\frac{q-1}{2}}}^T\right)^T$, $i = 1, \ldots, \binom{m}{q}$.
3. Generate design $D = (D_1^T, \ldots, D_{\binom{m}{q}}^T)^T$.

Theorem 6 *For any $m \geq 3$, prime $2 \leq q < m$, design D obtained by Algorithm 4 is D-, G-, and I-optimal under LCPS model (1), with run size $n = \frac{q(q-1)}{2}\binom{m}{q}$.*

Example 1 Suppose an order-of-addition experiment involves 6 components, but each run can accommodate at most 5 of them. That is, $m = 6$, $q = 5$. Now we construct an order-of-addition screening design by Algorithm 4.

1. we generate the set of all 5-component combinations as $\mathcal{A} = \{(1,2,3,4,5), (1,2,3,4,6), (1,2,\ 3,5,6), (1,2,4,\ 5,6), (1,3,4,5,6), (2,3,4,5,6)\}$.

2. For each α in $\mathcal{A}$, we construct 2 generating vectors and corresponding retrocirculant Latin square,

$$L_{(2,3,4,5,1)+}, L_{(2,3,4,6,1)+}, L_{(2,3,5,6,1)+}, L_{(2,4,5,6,1)+}, L_{(3,4,5,6,1)+}, L_{(3,4,5,6,2)+},$$
$$L_{(3,5,2,4,1)+}, L_{(3,6,2,4,1)+}, L_{(3,6,2,5,1)+}, L_{(4,6,2,5,1)+}, L_{(4,6,3,5,1)+}, L_{(4,6,3,5,2)+}.$$

And we define

$$D_1 : \begin{pmatrix} L_{(2,3,4,5,1)+} \\ L_{(3,5,2,4,1)+} \end{pmatrix}, D_2 : \begin{pmatrix} L_{(2,3,4,6,1)+} \\ L_{(3,6,2,4,1)+} \end{pmatrix}, D_3 : \begin{pmatrix} L_{(2,3,5,6,1)+} \\ L_{(3,6,2,5,1)+} \end{pmatrix},$$
$$D_4 : \begin{pmatrix} L_{(2,3,5,6,1)+} \\ L_{(4,6,2,5,1)+} \end{pmatrix}, D_5 : \begin{pmatrix} L_{(3,4,5,6,1)+} \\ L_{(4,6,3,5,1)+} \end{pmatrix}, D_6 : \begin{pmatrix} L_{(3,4,5,6,2)+} \\ L_{(4,6,3,5,2)+} \end{pmatrix}.$$

Table 2 An order-of-addition screening design D for 6-component and 5-position in Example 1.

D_1:(1,2,3,4,5)						D_2:(1,2,3,4,6)						D_3:(1,2,3,5,6)					
Run	a_1	a_2	a_3	a_4	a_5	Run	a_1	a_2	a_3	a_4	a_5	Run	a_1	a_2	a_3	a_4	a_5
1	**2**	**3**	**4**	**5**	**1**	11	**2**	**3**	**4**	**6**	**1**	21	**2**	**3**	**5**	**6**	**1**
2	3	4	5	1	2	12	3	4	6	1	2	22	3	5	6	1	2
3	4	5	1	2	3	13	4	6	1	2	3	23	5	6	1	2	3
4	5	1	2	3	4	14	6	1	2	3	4	24	6	1	2	3	5
5	1	2	3	4	5	15	1	2	3	4	6	25	1	2	3	5	6
6	**3**	**5**	**2**	**4**	**1**	16	**3**	**6**	**2**	**4**	**1**	26	**3**	**6**	**2**	**5**	**1**
7	5	2	4	1	3	17	6	2	4	1	3	27	6	2	5	1	3
8	2	4	1	3	5	18	2	4	1	3	6	28	2	5	1	3	6
9	4	1	3	5	2	19	4	1	3	6	2	29	5	1	3	6	2
10	1	3	5	2	4	20	1	3	6	2	4	30	1	3	6	2	5
D_4:(1,2,4,5,6)						D_5:(1,3,4,5,6)						D_6:(2,3,4,5,6)					
Run	a_1	a_2	a_3	a_4	a_5	Run	a_1	a_2	a_3	a_4	a_5	Run	a_1	a_2	a_3	a_4	a_5
31	**2**	**4**	**5**	**6**	**1**	41	**3**	**4**	**5**	**6**	**1**	51	**3**	**4**	**5**	**6**	**2**
32	4	5	6	1	2	42	4	5	6	1	3	52	4	5	6	2	3
33	5	6	1	2	4	43	5	6	1	3	4	53	5	6	2	3	4
34	6	1	2	4	5	44	6	1	3	4	5	54	6	2	3	4	5
35	1	2	4	5	6	45	1	3	4	5	6	55	2	3	4	5	6
36	**4**	**6**	**2**	**5**	**1**	46	**4**	**6**	**3**	**5**	**1**	56	**4**	**6**	**3**	**5**	**2**
37	6	2	5	1	4	47	6	3	5	1	4	57	6	3	5	2	4
38	2	5	1	4	6	48	3	5	1	4	6	58	3	5	2	4	6
39	5	1	4	6	2	49	5	1	4	6	3	59	5	2	4	6	3
40	1	4	6	2	5	50	1	4	6	3	5	60	2	4	6	3	5

3. We generate design $D = (D_1^T, \ldots, D_6^T)^T$ with 60 runs as in Table 2.

For design D given in Example 1, it achieves $D_{\text{l,eff}} = 1$ with run size only accounts for $60/720 = 8.33\%$ of the full design. This indicates that the design constructed by Algorithm 4 is both efficient and practical under the new model. While Stokes and Xu (2024) is unable to construct the corresponding optimal design under the CPS model for $m = 6$ and $q = 5$, the optimal design under the PWOS model proposed by Stokes and Xu (2024) requires 72 runs.

In this section, we describe the generation of optimal designs under proposed new model. When experimental budget is small, a saturated but efficient design can be considered. The construction of such designs will be the focus of the next section.

4 Efficient design construction

First, when the experimental budget is severely constrained, we propose Algorithm 5 to construct an efficient design under the LCPS model (1) . When a slightly more generous budget is available, Algorithm 6 can be employed to obtain a design suitable for fitting either QCPS model (2) or SCPS model (3). The choice between these two algorithms can be made flexibly depending on the specific experimental context.

Algorithm 5

1. Generate $\binom{m}{q}$ q-component combinations, denoted as set $\mathcal{A} = \{(i_1, i_2, \ldots, i_q),\ 1 \leq i_1 < \cdots < i_q \leq m\}$. Randomly select one element from $\mathcal{A}$, written as $(i_1, i_2, \ldots, i_q)$, with the remaining components designated as $(j_1, j_2, \ldots, j_{m-q})$.
2. Generate the rows sequentially. Specifically, the 1st run is $(i_1, i_2, \ldots, i_q)$. For the 2nd run, shift each element in 1st run one position to the left, and place j_1 in the last position to obtain $(i_2, i_3, \ldots, j_1)$. For the 3rd run, shift each element in 2nd one position to the left, and place j_2 in the last position to obtain $(i_3, \ldots, j_1, j_2)$. By repeating this operation $m-1$ times, an order-of-addition screening design for m-component and q-position can be generated, denoted by $L_{m,q}$, the form of which is given in Equation (7).

$$L_{m,q} = \begin{pmatrix} i_1 & i_2 & i_3 & \ldots & i_{q-1} & i_q \\ i_2 & i_3 & i_4 & \ldots & i_q & j_1 \\ i_3 & i_4 & i_5 & \ldots & j_1 & j_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ j_{m-q-1} & j_{m-q} & i_1 & \ldots & i_{q-3} & i_{q-2} \\ j_{m-q} & i_1 & i_2 & \ldots & i_{q-2} & i_{q-1} \end{pmatrix}. \tag{7}$$

3. Permute the columns of $L_{m,q}$ to enhance its performance under the chosen criterion.

The design consists of m experimental runs, thereby constituting a saturated design for the LCPS model (1). Notably, it enables the fitting of proposed LCPS model (1), which cannot be accomplished by existing models or methods. For better understanding, Example 2 is employed to illustrate Algorithm 5.

Example 2 Consider an order-of-addition screening experiment involving 6 components but 4 positions. That is, $m = 6$, $q = 4$. We will construct a design with only 6 runs by Algorithm 5.

First, we obtain all possible combinations of 4 components, denoted as set $\mathcal{A}$ in Equation (8),

$$\{(1,2,3,4),(1,2,3,5),(1,2,3,6),(1,2,4,5),(1,2,4,6),(1,2,5,6),(1,3,4,5),(1,3,4,6),(1,3,5,6),(1,4,5,6),(2,3,4,5),(2,3,4,6),(2,3,5,6),(2,4,5,6),(3,4,5,6)\}. \tag{8}$$

A combination $(1,2,3,6)$ is randomly selected from $\mathcal{A}$, and the remaining components are denoted as $(4,5)$.

Next, we generate the rows sequentially: $(1,2,3,6)$ serves as the first row, $(2,3,6,4)$ constitutes the second row, and so on by step 2. As a result, we can obtain $L_{6,4}$ as shown in the left-hand side of Equation (9).

Finally, we use the D-criterion to select a column permutation. An exhaustive search over the column permutations of $L_{6,4}$ gives the best permutation, denoted by $L_{6,4}^{D\text{-}perm}$, on the right-hand side of Equation (9); it has $D_{\text{l,eff}} = 0.9881$ and $n_{L_{6,4}^{D\text{-}perm}}/n_{\text{full}} = 6/360 = 0.0167$. This statement concerns optimality within the specified class of column permutations, rather than global D-optimality over all six-run screening designs.

$$L_{6,4} = \begin{pmatrix} 1 & 2 & 3 & 6 & 4 & 5 \\ 2 & 3 & 6 & 4 & 5 & 1 \\ 3 & 6 & 4 & 5 & 1 & 2 \\ 6 & 4 & 5 & 1 & 2 & 3 \end{pmatrix}^T, \quad L_{6,4}^{D\text{-}perm} = \begin{pmatrix} 3 & 6 & 4 & 5 & 1 & 2 \\ 1 & 2 & 3 & 6 & 4 & 5 \\ 6 & 4 & 5 & 1 & 2 & 3 \\ 2 & 3 & 6 & 4 & 5 & 1 \end{pmatrix}^T. \tag{9}$$

Remark 3

1. Regarding the selection of the initial vector in step 1 of Algorithm 5, we set it to be random. In practical applications, multiple sets of initial vectors can be used to generate designs, from which the one with the highest efficiency is selected.
2. m is the minimum number of runs to ensure identifiability of all parameters in LCPS model for any m and q; this is attributed to the constraint $\sum_{k=1}^{m} b_k = \frac{q(q+1)}{2}$, and the number of free parameters in LCPS model is m.

Algorithm 5 is suitable for scenarios with a highly limited experimental budget. When the budget is slightly increased, Algorithm 6 and the QCPS model (2), SCPS model (3) can be adopted instead. The previous construction approach remains applicable, with only a few more $L_{m,q}$ to be selected. The specific process is detailed in Algorithm 6.

Remark 4

1. The selection of t can be flexibly determined based on the experimental budget and the number of model parameters, with the aim of choosing the largest possible value for t.
2. The stopping rule in step 5 must be specified before the search; it can be a target D-efficiency, a fixed number of starting designs, or a fixed computing-time budget. The random seed and the attained efficiency should be reported.
3. If $q = 2$, we do not recommend using Algorithm 6 to construct the order-of-addition screening design under the SCPS model, as its model matrix tends to contain a large number of 0.

Algorithm 6

1. Generate $\binom{m}{q}$ q-component combinations to form set $\mathcal{A} = \{(i_1, i_2, \ldots, i_q), 1 \le i_1 < i_2 < \ldots i_q \le m\}$.
2. Select t elements from $\mathcal{A}$ within budget such that the chosen model can be estimated, and generate the corresponding designs by step 2 in Algorithm 5, denoted as $L^1_{m,q}, \ldots, L^t_{m,q}$.
3. Obtain L by juxtaposing the rows of $L^i_{m.q}$, $i = 1, \ldots, t$, i.e. $L = (L^{1T}_{m,q}, \ldots, L^{tT}_{m,q})^T$.
4. Permute L's columns to enhance its performance under a given criterion.
5. Repeat steps 1–4 according to a pre-specified search budget or until a pre-specified efficiency threshold is reached, and retain the best-found design under the chosen criterion.

Example 3 (**Example 2-continued**) Suppose the experiment settings are the same as Example 2. We consider constructing suitable designs based on Algorithm 6 for the QCPS model (2) and SCPS model (3), respectively.

As the the QCPS model (2) with 11 parameters, we begin by taking 2 elements from $\mathcal{A}$ in Equation (8), and generate the corresponding designs, denoted as $L^1_{6,4}$ and $L^2_{6,4}$. By concatenating their rows together as described in step 3, we get the design L_Q, as in Equation (10). Then, we apply column permutations to L_Q. Using 105 candidate constructions and the stated column-permutation search, we obtain the best-found QCPS design $L^\star_Q$, shown in Equation (10).

Similarly, for the SCPS model (3), there are 20 parameters to be estimated. By Algorithm 6, we select 4 elements from $\mathcal{A}$ and generate the corresponding designs, denoted as $L^1_{6,4}$, $L^2_{6,4}$, $L^3_{6,4}$, and $L^4_{6,4}$. By concatenating their rows, we obtain L_S as shown in Equation (11). Next, we permute the columns of L_S and search 1365 candidate constructions. The best-found SCPS design, denoted by $L^\star_S$, is provided in Equation (12).

$$
\begin{aligned}
L_Q &= \left({L^1_{6,4}}^T, {L^2_{6,4}}^T\right)^T = \left(\begin{array}{cccccc|cccccc} 2&3&4&5&1&6&1&3&5&6&2&4\\ 3&4&5&1&6&2&3&5&6&2&4&1\\ 4&5&1&6&2&3&5&6&2&4&1&3\\ 5&1&6&2&3&4&6&2&4&1&3&5 \end{array}\right)^T,\\
L^\star_Q &= \left(\begin{array}{cccccc|cccccc} 2&3&4&5&1&6&1&3&5&6&2&4\\ 4&5&1&6&2&3&5&6&2&4&1&3\\ 5&1&6&2&3&4&6&2&4&1&3&5\\ 3&4&5&1&6&2&3&5&6&2&4&1 \end{array}\right)^T.
\end{aligned} \tag{10}
$$

$$
\begin{aligned}
L_S &= \left({L^1_{6,4}}^T, {L^2_{6,4}}^T, {L^3_{6,4}}^T, {L^4_{6,4}}^T\right)^T\\
&= \left(\begin{array}{cccccc|cccccc|cccccc|cccccc} 2&3&4&6&1&5&1&4&5&6&2&3&1&2&4&5&3&6&1&2&3&4&5&6\\ 3&4&6&1&5&2&4&5&6&2&3&1&2&4&5&3&6&1&2&3&4&5&6&1\\ 4&6&1&5&2&3&5&6&2&3&1&4&4&5&3&6&1&2&3&4&5&6&1&2\\ 6&1&5&2&3&4&6&2&3&1&4&5&5&3&6&1&2&4&4&5&6&1&2&3 \end{array}\right)^T.
\end{aligned} \tag{11}
$$

$$
L^\star_S = \left(\begin{array}{cccccc|cccccc|cccccc|cccccc} 6&1&5&2&3&4&6&2&3&1&4&5&5&3&6&1&2&4&4&5&6&1&2&3\\ 4&6&1&5&2&3&5&6&2&3&1&4&4&5&3&6&1&2&3&4&5&6&1&2\\ 3&4&6&1&5&2&4&5&6&2&3&1&2&4&5&3&6&1&2&3&4&5&6&1\\ 2&3&4&6&1&5&1&4&5&6&2&3&1&2&4&5&3&6&1&2&3&4&5&6 \end{array}\right)^T. \tag{12}
$$

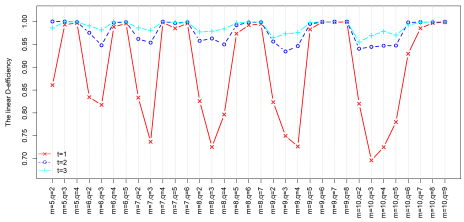


Fig. 1 The $D_{\text{l,eff}}$ for different m, q and t.

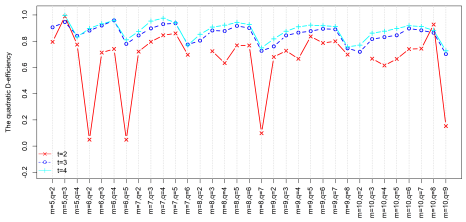


Fig. 2 The $D_{\text{q,eff}}$ for different m, q and t.

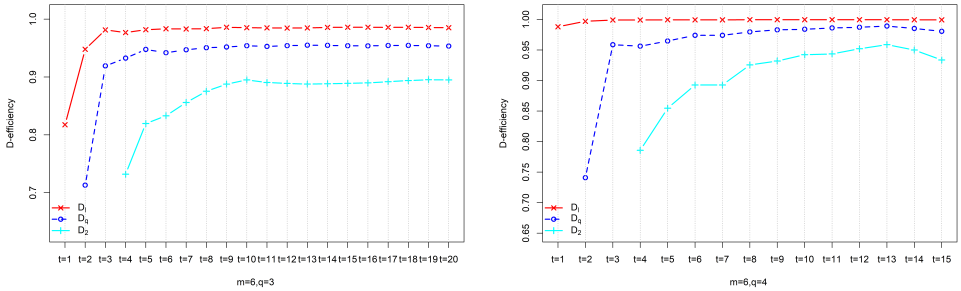


Fig. 3 The D-efficiency for different t under $m = 6$, $q = 3$ (left), and $m = 6$, $q = 4$ (right).

In Table B1–B3 (Appendix B), we list the D-efficiencies and run size of order-of-addition screening designs generated by Algorithm 5–6 with different m, q, and t. The "–" indicates that the corresponding design is not required to exist, due to the number of runs is the same as that of the full design, or the corresponding D-efficiency does not exist. From these tables,

(a) The $D_{\text{l,eff}}$ values of $L_{m,q}$ generated by Algorithm 5 ($t = 1$) are generally high. Figure 1 provides a compact summary; most values exceed 0.75.

(b) The $D_{\text{q,eff}}$ and $D_{\text{s,eff}}$ values of the best-found designs generated by Algorithm 6 ($t \geq 2$) are given in Tables B1–B3 in Appendix B. Figure 2 shows selected $D_{\text{q,eff}}$ values. As expected, the quadratic efficiencies for $t = 3$ and 4 generally exceed those for $t = 2$.

The value of t should be chosen from the experimental budget and the required model complexity. For example, when $m = 10$, $q = 3$, increasing from $t = 1$ to $t = 2$ raises $D_{\text{l,eff}}$ from 0.6963 to 0.9445, a 35.65% increase. Thus, LCPS can use $L_{m,q}$ from Algorithm 5 under a severely limited budget; QCPS and SCPS should use Algorithm 6 with the largest feasible t that still permits the intended model to be estimated.

5 Numerical simulations

This section evaluates the practical value of the proposed designs beyond D-efficiency. We compare a design generated by Algorithm 6 with an equal-budget simple random screening design. In both cases the response is analysed using the model targeted by the design. For every replicate, prediction is made over the complete candidate set. We report standardised mean squared prediction error (SMSPE), the probabilities that the predicted best sequence is the true best sequence (Top-1) or belongs to the true top five sequences (Top-5), and the decision regret $y(\widehat{x}) - \min_x y(x)$. Each entry is the Monte Carlo mean over 300 independent response vectors; the Monte Carlo standard error is given in parentheses. The error standard deviation is 0.10 times the standard deviation of the noise-free response over the candidate set.

5.1 Constrained job scheduling

We first consider the constrained job-scheduling setting of Stokes and Xu (2024). Exactly $q = 4$ of $m = 11$ jobs are processed in each run, so the candidate set contains $11!/(11-4)! = 7{,}920$ screened sequences. For $x = (x_1, \ldots, x_q)$, the total cost is

$$y(x) = \sum_{i=1}^{q} c_{x_i} \left(\sum_{j=1}^{i} t_{x_j} \right)^2 + \sum_{k \notin x} p_k + \varepsilon. \tag{13}$$

The fixed vectors of processing times, costs, and omission penalties are, respectively,

$$\begin{aligned} t &= (8, 16, 10, 9, 12, 14, 6, 15, 11, 7, 13), \\ c &= (16, 5, 12, 13, 9, 7, 18, 11, 6, 15, 10), \\ p &= (107, 98, 110, 89, 96, 101, 93, 112, 87, 104, 99). \end{aligned}$$

We use a $33 = 3m$ run QCPS design. QCPS has 21 free parameters. CPS and PWOS require 41 and 56 parameters, respectively, and are not identifiable without regularisation at this budget.

Table 3 Job-scheduling performance for $(m, q, n) = (11, 4, 33)$. Values in parentheses are Monte Carlo standard errors.

Design	SMSPE	Top-1	Top-5	Regret
Proposed QCPS	0.1581 (0.0011)	0.007 (0.005)	0.953 (0.012)	1353.16 (19.70)
Random QCPS	0.3285 (0.0173)	0.227 (0.024)	0.403 (0.028)	2398.81 (131.62)

5.2 Constrained multi-drug pulse screening

To assess a setting with nonlinear position effects and a small number of interactions, we consider eight candidate drugs, of which exactly five can be administered in one

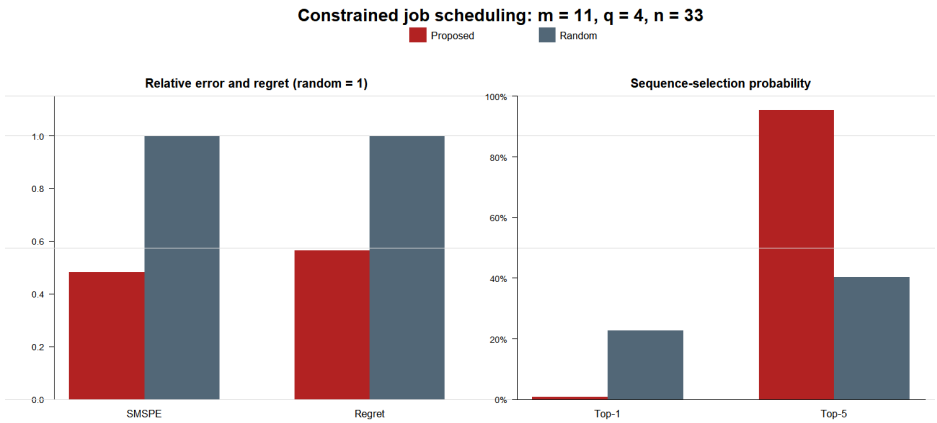


Fig. 4 Job-scheduling simulation. The left panel reports SMSPE and regret relative to random screening; the right panel reports the Top-1 and Top-5 selection probabilities.

treatment cycle. This is a screening extension of the multi-drug sequencing motivation in Stokes and Xu (2022); it contains $8!/(8-5)! = 6{,}720$ admissible sequences. The noise-free response follows a sparse SCPS model with four nonzero linear effects, three nonzero quadratic effects, and three nonzero two-drug interactions. The coefficient magnitudes are $(-4, 3, -2.5, 2)$ for the linear terms, $(1.1, -0.9, 0.7)$ for the quadratic terms, and $(1.2, -1, 0.8)$ for the interactions. We fit SCPS using a $48 = 6m$ run design, which has 35 free parameters. This scenario directly tests whether the proposed construction retains predictive and decision value when position effects are nonlinear and partly interactive.

Table 3 and Figure 4 show that the proposed QCPS design nearly halves SMSPE and reduces mean regret by 43.6% relative to random screening. Random screening has the larger exact Top-1 rate in this particular configuration; we report this result rather than suppressing an unfavourable endpoint. However, the much larger Top-5 probability and lower regret show that QCPS more reliably identifies practically competitive schedules.

Table 4 Multi-drug pulse-screening performance for $(m, q, n) = (8, 5, 48)$. Values in parentheses are Monte Carlo standard errors.

Design	SMSPE	Top-1	Top-5	Regret
Proposed SCPS	0.0311 (0.0009)	0.390 (0.028)	0.867 (0.020)	1.003 (0.092)
Random SCPS	0.0338 (0.0012)	0.323 (0.027)	0.780 (0.024)	1.358 (0.106)

Table 4 and Figure 5 show that the proposed SCPS design improves all four criteria, including a 26.1% reduction in regret. Together, the two scenarios show that high information efficiency can translate into useful prediction and sequence-selection gains, particularly in the nonlinear interaction setting for which SCPS was introduced.

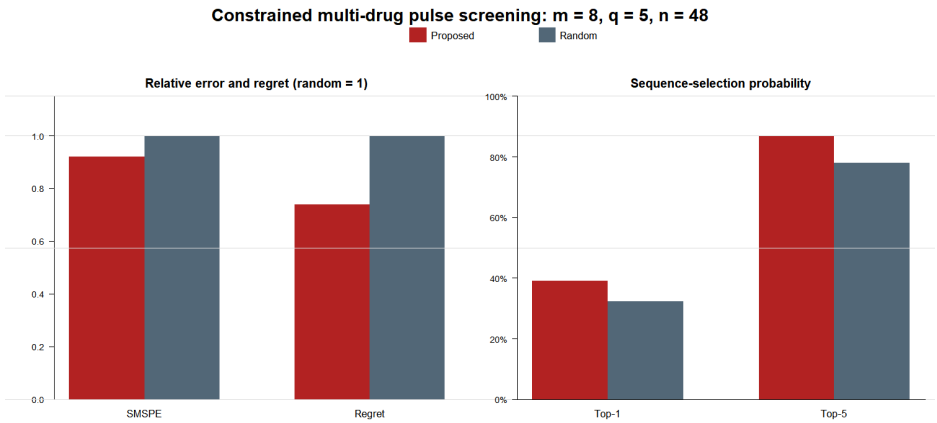


Fig. 5 Multi-drug pulse-screening simulation. The proposed SCPS design improves prediction, selection, and regret relative to equal-budget random screening.

6 Discussion

This paper introduces the LCPS, QCPS, and SCPS models for order-of-addition screening experiments. These models treat omission explicitly through position zero and allow the model complexity to be aligned with the experimental budget. We establish optimality results for the full screening design, show that several existing constructions retain their optimality, and give new exact or high-efficiency constructions for general settings.

The numerical results show that the value of the proposed framework is not limited to D-efficiency. In the constrained scheduling experiment, QCPS produces substantially more accurate predictions and lower regret than an equal-budget random screening design, even when CPS and PWOS are not estimable without regularisation. In the constrained multi-drug pulse experiment, SCPS improves both prediction and the identification of competitive treatment sequences under nonlinear and interactive position effects. Future work should develop constructions targeted directly at SCPS prediction and sequential decision criteria, and should characterise the robustness of the proposed designs under mixtures of absolute- and relative-position effects.

Declarations

The authors declare that they have no conflict of interest.

References

Chen J, Mukerjee R, and Lin D K, Construction of optimal fractional order-of-addition designs via block designs, *Statistics & Probability Letters*, 2020, **161**: 108728.

Hedayat A S, Sloane N J A, and Stufken J, *Orthogonal Arrays: Theory and Applications*, Springer Science & Business Media, 1999.

Huang H, Construction of component orthogonal arrays with any number of components, *Journal of Statistical Planning and Inference*, 2021, **213**: 72–79.

Peng J, Mukerjee R, and Lin D K, Design of order-of-addition experiments, *Biometrika*, 2019, **106**(3): 683–694.

Schoen E D and Mee R W, Order-of-addition orthogonal arrays to study the effect of treatment ordering, *The Annals of Statistics*, 2023, **51**(4): 1877–1894.

Stokes Z and Xu H, A position-based approach for design and analysis of order-of-addition experiments, *Statistica Sinica*, 2022, **32**(3), 1467–1488.

Stokes Z and Xu H, Designs for order-of-addition screening experiments, *Statistica Sinica*, 2024, **34**(1): 399–419.

Tsai S F, Generating optimal order-of-addition designs with flexible run sizes, *Journal of Statistical Planning and Inference*, 2022, **218**: 147–163.

Van Nostrand R C, Design of experiments where the order of addition is important, In *ASA proceedings of the Section on Physical and Engineering Sciences* (Vol. 155, pp. 160), Alexandria, VA: American Statistical Association, 1995.

Voelkel J G, The design of order-of-addition experiments, *Journal of Quality Technology*, 2019, **51**(3), 230–241.

Wang C, King C, and Lin D K, A general construction method for component orthogonal arrays, *Quality and Reliability Engineering International*, 2024, **40**(1), 712–721.

Yang J F, Sun F, and Xu H, A component-position model, analysis and design for order-of-addition experiments, *Technometrics*, 2021, **63**(2), 212–224.

Zhang X, *Matrix analysis and applications*, Cambridge University Press, 2017.

Zhao Y, Li Z, and Zhao S, A new method of finding component orthogonal arrays for order-of-addition experiments, *Metrika*, 2021a, **84**(6), 805–824.

Zhao Y, Lin D K, and Liu M Q, Designs for order-of-addition experiments. *Journal of Applied Statistics*, 2021b, **48**(8), 1475–1495.

Appendix A Proofs of the theorems

The following lemma is needed to prove the theorems.

Lemma 1 Assume $g \neq 0$, $f \neq 0$, and $gk + f \neq 0$, then

(i) $|gI_k + fJ_k| = g^{k-1}(g + fk)$;

(ii) $(gI_k + fJ_k)^{-1} = \frac{1}{g}I_k - \frac{f}{g(fk+g)}J_k$,

where I_k denotes the $k \times k$ identity matrix, and J_k represents the $k \times k$ matrix with all elements equal to 1.

Lemma 2 (Zhang, 2017) When the matrix A is invertible, we have

$$\begin{pmatrix} A & U \\ V & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} + A^{-1}U\left(D - VA^{-1}U\right)^{-1}VA^{-1} & -A^{-1}U\left(D - VA^{-1}U\right)^{-1} \\ -\left(D - VA^{-1}U\right)^{-1}VA^{-1} & \left(D - VA^{-1}U\right)^{-1} \end{pmatrix}.$$

Proof (Proof of Theorem 1) For $m \geq 3$ and $1 < q < m$, the model matrix X_l for the LCPS model (4) can be expressed as $X_l = (\mathbf{1}, B_l)$. The normalized information matrix and its inverse for the full design $S_{m,q}$ under the LCPS model are given by

$$M(S_{m,q}) = \frac{X_l^T X_l}{N} = \begin{pmatrix} \mathbf{1}^T\mathbf{1} & \mathbf{1}^T B_l \\ B_l^T\mathbf{1} & B_l^T B_l \end{pmatrix} = \begin{pmatrix} 1 & c\mathbf{1}^T \\ c\mathbf{1} & (a-b)I + bJ \end{pmatrix}, \tag{A1}$$

$$M^{-1}(S_{m,q}) = \frac{1}{a-b}\begin{pmatrix} d & e\mathbf{1}^T \\ e\mathbf{1} & I + J \end{pmatrix}, \tag{A2}$$

where

$$N = \binom{m}{q}q!,\ a = \frac{q(q+1)(2q+1)}{6m},\ b = \frac{q(3q+2)(q^2-1)}{12m(m-1)},$$
$$c = \frac{q(q+1)}{2m},\ d = a - b + \frac{q^2(q+1)^2(m-1)}{4m},\ e = -\frac{q(q+1)}{2}.$$

In order to apply the equivalence theorem, we need to calculate whether the sensitivity function $X_i M^{-1}(S_{m,q})X_i^T - m$ is equal to 0 for each i, where $X_i = (1, b_{i1}, b_{i2}, \ldots, b_{i(m-1)})$ represents the ith row of model matrix X_l. Let b_{im} be the position of component m in the ith run, then obviously $b_{i1} + \cdots + b_{i(m-1)} + b_{im} = \frac{q(q+1)}{2}$. We have

$$\begin{aligned} & X_i M^{-1}(S_{m,q})X_i^T \\ =& \frac{1}{a-b}\left[d + 2e(b_{i1} + \cdots + b_{i(m-1)}) + (b_{i1} + \cdots + b_{i(m-1)})^2 + b_{i1}^2 + \cdots + b_{i(m-1)}^2\right] \\ =& \frac{1}{a-b}\left[d - (e - b_{im})(-e - b_{im}) + \left(\frac{q(q+1)(2q+1)}{6} - b_{im}^2\right)\right], \end{aligned}$$

which is equal to m after simple calculations. Therefore, by the equivalence theorem, the full design $S_{m,q}$ is D-optimal and G-optimal under the LCPS model (1). In addition, Schoen and Mee (2023) pointed out that for the uniform design measure in a discrete design space, G-optimality is equivalent to I-optimality. Therefore, the full design $S_{m,q}$ is also I-optimal under the LCPS model (1).

Therefore, the proof of Theorem 1 is completed. □

Proof (Proof of Theorem 2) For $m \geq 3$ and $1 < q < m$, the model matrix X_q for the QCPS model (5) can be expressed as

$$X_q = (\mathbf{1}, B_l, B_q).$$

The normalized information matrix for the full design $S_{m,q}$ under the QCPS model (2) is computed as

$$\begin{aligned} M =& \frac{X_q^T X_q}{N} = \frac{1}{N}\begin{pmatrix} \mathbf{1}^T\mathbf{1} & \mathbf{1}^T B_l & \mathbf{1}^T B_q \\ B_l^T\mathbf{1} & B_l^T B_l & B_l^T B_q \\ B_q^T\mathbf{1} & B_q^T B_l & B_q^T B_q \end{pmatrix} \\ =& \begin{pmatrix} 1 & c\mathbf{1}^T & a\mathbf{1}^T \\ c\mathbf{1} & (a-b)I + bJ & (u-g)I + gJ \\ a\mathbf{1} & (u-g)I + gJ & (f-r)I + rJ \end{pmatrix}, \end{aligned} \tag{A3}$$

where

$$\begin{aligned} f &= \frac{q(q+1)(2q+1)(3q^2+3q-1)}{30m},\ u = \frac{q^2(q+1)^2}{4m}, \\ r &= \frac{q(q+1)(2q+1)(10q^3-3q^2-13q+6)}{180m(m-1)},\ g = \frac{q^2(q+1)^2(q-1)}{6m(m-1)}, \end{aligned}$$

the remaining a, b, and c are the same as the proof of Theorem 1. Using Lemma 1 and 2, we obtain

$$M^{-1} = \frac{1}{l}\begin{pmatrix} v & w\mathbf{1}^T & h\mathbf{1}^T \\ w\mathbf{1} & (y-z)I + zJ & (j-k)I + kJ \\ h\mathbf{1} & (j-k)I + kJ & (s-t)I + tJ \end{pmatrix}, \tag{A4}$$

where

$$\begin{aligned} &l = 3m(3q^2+3q+2) - 4q(q+1)(2q+1), \\ &v = m(8mq^3 - 16q^3 + 12mq^2 - 15q^2 + 4mq + q + 6), \\ &w = -18m(m-1)(2q+1),\ h = 30m(m-1), \\ &z = \frac{12(m-1)(2q+1)(-10q^3+18mq^2-15q^2+18mq-5q-6m)}{q(q^2-1)(q+2)}, \\ &k = \frac{-180(m-1)(3m-2q-1)}{(q-1)(q+2)},\ t = \frac{180(m-1)(-3q^2+4mq-3q+2m)}{q(q^2-1)(q+2)}, \\ &y = 2z,\ j = 2k,\ s = 2t. \end{aligned}$$

To apply the equivalence theorem, we need to calculate the sensitivity function $X_i M^{-1} X_i^T$ for each i, where $X_i = (1, b_{i1}, b_{i2}, \ldots, b_{i(m-1)}, b_{i1}^2, b_{i2}^2, \ldots, b_{i(m-1)}^2)$ represents the ith row of model matrix X. Let b_{im} be the position of component m in the ith run, then the results below are obvious,

$$\begin{aligned} b_{i1} + \cdots + b_{i(m-1)} + b_{im} &= \frac{q(q+1)}{2}, \\ b_{i1}^2 + \cdots + b_{i(m-1)}^2 + b_{im}^2 &= \frac{q(q+1)(2q+1)}{6}, \\ b_{i1}^3 + \cdots + b_{i(m-1)}^3 + b_{im}^3 &= \frac{q^2(q+1)^2}{4}, \\ b_{i1}^4 + \cdots + b_{i(m-1)}^4 + b_{im}^4 &= \frac{q(q+1)(2q+1)(3q^2+3q-1)}{30}. \end{aligned}$$

Using these results, we have

$$\begin{aligned} X_i M^{-1} X_i^T =& b_m^2\left[-2h - \frac{jq(q+1)}{2} - \frac{tq(q+1)(2q+1)}{3}\right] \\ &+ b_m\left[-2w - zq(q+1) - \frac{jq(q+1)(2q+1)}{6}\right] \end{aligned}$$

$$+ v + wq(q+1) + \frac{q(q+1)(2q+1)(2h+z)}{6} + \frac{(j+z)q^2(q+1)^2}{4}$$
$$+ \frac{tq(q+1)(2q+1)(3q^2+3q-1)}{30} + \frac{jq^2(q+1)^2(2q+1)}{12}$$
$$+ \frac{tq^2(q+1)^2(2q+1)^2}{36},$$

which is equal to $2m-1$ after a series of simplifications. Therefore, by the equivalence theorem, the full design $S_{m,q}$ is D-optimal and G-optimal under the QCPS model (2). At the same time, using Theorem 2 in Schoen and Mee (2023), we know that the full design $S_{m,q}$ is also I-optimal under the QCPS model (2).

Therefore, the proof of Theorem 2 is completed. □

Proof (Proof of Theorem 3 and Remark 1) Let m be a prime or a prime power, $1 < q < m$, and $n = \lambda m(m-1)$ with $\lambda \geq 1$. The design $S^{CP}_{n,m,q}$ inherits the properties of $COA(n,m)$, that is, every level combination (i,j) with $i \neq j$, $i,j = 1,\ldots,m$ appears equally for any two-column sub-array. After converting the design $S^{CP}_{n,m,q}$ into a position matrix B, we have

(1) for each column b_k in B with $k = 1,\ldots,m-1$, each level $1,\ldots,q$ appears $\lambda(m-1)$ times, and level 0 appears $\lambda(m-1)(m-q)$ times;
(2) for any two columns b_k and $b_k^{'}$ in B with $k \neq k'$, $k,k' = 1,\ldots,m-1$, each level combination (i,j) with $i \neq j$, $i,j = 1,\ldots,q$ appears λ times, and level combination (i,j) with $i = 0$ or $j = 0$ appears $\lambda(m-q)(m+q-1)$ times.

Thus, we have

$$\frac{\mathbf{1}_n^T b_k}{n} = \frac{q(q+1)}{2m},$$
$$\frac{\mathbf{1}_n^T b_k^2}{n} = \frac{q(q+1)(2q+1)}{6m},$$
$$\frac{b_k^T b_k}{n} = \frac{q(q+1)(2q+1)}{6m},$$
$$\frac{b_k^T b_{k'}}{n} = \frac{q(q^2-1)(3q+2)}{12m(m-1)},$$
$$\frac{b_k^T b_k^2}{n} = \frac{q^2(q+1)^2}{4m},$$
$$\frac{b_k^T b_{k'}^2}{n} = \frac{q^2(q+1)^2(q-1)}{6m(m-1)},$$
$$\frac{b_k^{2T} b_k^2}{n} = \frac{q(q+1)(2q+1)(3q^2+3q-1)}{30m},$$
$$\frac{b_k^{2T} b_{k'}^2}{n} = \frac{q(q+1)(2q+1)(10q^3-3q^2-13q+6)}{180m(m-1)}.$$

Therefore, the normalized information matrices of design $S^{CP}_{n,m,q}$ under the LCPS model (1) and QCPS model (2) are the same as the full design $S_{m,q}$ given by Theorem 1-2. The design $S^{CP}_{n,m,q}$ is D-, G-, and I-optimal under the LCPS model (1) and QCPS model (2) .

Therefore, the proof of Theorem 3 and Remark 1 is completed. □

Proof (Proof of Theorem 4) To prove the optimality of design $S^{PWO}_{n,m,3}$, we only need to demonstrate that its normalized information matrix is the same as the full design. For given design $S^{PWO}_{n,m,3}$ with $n = 3\binom{m}{3}$ and m even, we denote its model matrix under the LCPS model (1) as $X_l = (\mathbf{1}_n, b_1, \ldots, b_{m-1})$. The main elements in its normalized information matrix including $\mathbf{1}_n^T b_k/n$, $b_k^T b_k/n$, and $b_k^T b_{k'}/n$, $k \neq k'$, $k, k' = 1, \ldots, m-1$. Next we calculate them respectively.

First, we calculate $\mathbf{1}_n^T b_k/n$ for $k = 1, \ldots, m-1$. Based on the design structure, there are $\binom{m-1}{2}$ design blocks $D'_{klj}s$ containing component k, $k \in \{1, \ldots, m-1\}$, where each D_{klj} ignoring the order of rows is either

$$\begin{pmatrix} k & j & l \\ l & k & j \\ j & l & k \end{pmatrix} \text{ or } \begin{pmatrix} k & l & j \\ l & j & k \\ j & k & l \end{pmatrix}.$$

For both types of D_{klj}, the value of b_k and its frequency are

$$\begin{array}{cc} \text{value} & 1\ 2\ 3; \\ \text{frequency} & 1\ 1\ 1. \end{array}$$

There are additionally $\binom{m-1}{3}$ design blocks $D_{k'lj}$ that do not contain component k. Thus, for the whole design $S^{PWO}_{n,m,3}$, the value of b_k and its frequency are as follows,

$$\begin{array}{cccccc} \text{value} & 1 & 2 & 3 & 0; \\ \text{frequency} & \binom{m-1}{2} & \binom{m-1}{2} & \binom{m-1}{2} & 3\binom{m-1}{3}. \end{array}$$

Therefore, we have

$$\frac{\mathbf{1}_n^T b_k}{n} = \frac{6}{m}, \ \frac{b_k^T b_k}{n} = \frac{14}{m}.$$

The calculation of $b_k^T b_k/n$ and $b_k^T b_{k'}/n$, $k \neq k'$, $k, k' = 1, \ldots, m-1$, are as follows. There are $\binom{m-2}{1}$ $D_{kk'j}$'s, where k and $k^{'}$ occur together. For both two types of $D_{kk'j}$, the value of b_k and $b_{k'}$ are

$$\begin{array}{cc} b_k & b_{k'} \\ \begin{pmatrix} 1 & 3 \\ 2 & 1 \\ 3 & 2 \end{pmatrix} & \text{or} \end{array} \quad \begin{array}{cc} b_k & b_{k'} \\ \begin{pmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 1 \end{pmatrix}, \end{array}$$

where each pair $(1,2), (1,3)$ and $(2,3)$ appears once, ignoring the order. Thus, we have

$$\frac{b_k^T b_{k'}}{n} = \frac{22}{m(m-1)}.$$

We can conclude that the information matrix of design $S^{PWO}_{n,m,3}$ is the same as the full design. The design $S^{PWO}_{n,m,3}$ is the D-, G-, and I-optimal under the LCPS model (1). Therefore, the proof of Theorem 4 is completed. □

Proof (Proof of Theorem 6) For design D obtained by Algorithm 6, we denote its model matrix under the LCPS model (1) as $X_l = (\mathbf{1}_n, B_l)$, where $n = \binom{m}{q}\frac{q(q-1)}{2}$, and $B_l = (b_1, \ldots, b_{m-1})$. Now in order to prove that design D is D-, G-, and I-optimal under the LCPS model (1), we only need to demonstrate that $\frac{X_l^T X_l}{n}$ is the same as the normalized information matrix of the full design.

We first consider one of the $D_i = \left(L_{\alpha_1}^T, \ldots, L_{\alpha_{\frac{q-1}{2}}}^T\right)^T$ with the corresponding q-component combination $l_0, \ldots, l_{q-1}$ and the position matrix $B_i = (b_1, \ldots, b_m)$, $i = 1, \ldots, \binom{m}{q}$. Since q is an odd prime, each $\alpha_1, \ldots, \alpha_{\frac{q-1}{2}}$ is a permutation of $l_0, \ldots, l_{q-1}$. Also based on the structure of D_i, we know that

(1) for each column $b_{l_0}, \ldots, b_{l_{q-1}}$, its value and frequency are

$$\begin{array}{lccccc} \text{value} & 1 & 2 & \cdots & q; \\ \text{times} & \frac{q-1}{2} & \frac{q-1}{2} & \ldots & \frac{q-1}{2}; \\ \text{frequency} & \frac{1}{q} & \frac{1}{q} & \ldots & \frac{1}{q}. \end{array}$$

(2) for each pair columns b_{l_j} and $b_{l_{j^*}}$, $l_j \neq l_{j^*}$, $l_j, l_{j^*} \in \{l_0, \ldots, l_{q-1}\}$, the value (ignoring the order) and their frequency are

$$\begin{array}{lccccccc} \text{value} & 12 & \ldots & 1q & 23 & \ldots & 2q & \ldots\ (q-1)q; \\ \text{times} & 1 & \ldots & 1 & 1 & \ldots & 1 & \ldots\ 1; \\ \text{frequency} & \frac{2}{q(q-1)} & \cdots & \frac{2}{q(q-1)} & \frac{2}{q(q-1)} & \cdots & \frac{2}{q(q-1)} & \cdots\ \frac{2}{q(q-1)}. \end{array}$$

For all the D_i's, there are $\binom{m-1}{q-1}$ of them containing the component k, and $\binom{m-2}{q-2}$ of them containing both components k and $k^{'}$, $k \neq k^{'}$, $k = 1, \ldots, m-1$. Then we have the following conclusions for design D,

(1) for column b_k, its value and the corresponding frequency are

$$\begin{array}{lccccc} \text{value} & 1 & 2 & \cdots & q & 0; \\ \text{times} & \frac{q-1}{2}\binom{m-1}{q-1} & \frac{q-1}{2}\binom{m-1}{q-1} & \ldots & \frac{q-1}{2}\binom{m-1}{q-1} & \frac{q(q-1)}{2}\binom{m-1}{q}; \\ \text{frequency} & \frac{1}{m} & \frac{1}{m} & \ldots & \frac{1}{m} & \frac{m-q}{m}. \end{array}$$

(2) for each pair columns b_k and $b_{k'}$, their value (ignoring the order) and frequency are

$$\begin{array}{lccccccc} \text{value} & 12 & \ldots & 1q & 23 & \ldots & (q-1)q & 0x/x0/00; \\ \text{times} & \binom{m-2}{q-2} & \ldots & \binom{m-2}{q-2} & \binom{m-2}{q-2} & \ldots & \binom{m-2}{q-2} & 2\binom{m-2}{q-1}\frac{q(q-1)}{2} + \binom{m-2}{q}\frac{q(q-1)}{2}; \\ \text{frequency} & \frac{2}{m(m-1)} & \cdots & \frac{2}{m(m-1)} & \frac{2}{m(m-1)} & \cdots & \frac{2}{m(m-1)} & \frac{(m-q)(m+q-1)}{m(m-1)}; \end{array}$$

where $x = 1, \ldots, q$.

Then, we can draw the following conclusions by calculation,

$$\mathbf{1}^T b_k = \frac{q(q-1)}{2m}, b_k^T b_k = \frac{q(q+1)(2q+1)}{6m}, b_k^T b_{k'} = \frac{q(q^2-1)(3q+1)}{12m(m-1)},$$

where $k \neq k'$, $k = 1, \ldots, m-1$. Therefore, under the LCPS model (1), the normalized information matrix of design D is the same as the full design. Design D from Algorithm 6 is D-, G- and I-optimal.

Therefore, the proof of Theorem 6 is completed. □

Appendix B D-efficiencies for different m, q, and t

Table B1 D-efficiencies for different $m \in \{3, 4, 5, 6, 7\}$, q, and t.

m		3	4		5			6				7				
q		2	2	3	2	3	4	2	3	4	5	2	3	4	5	6
run_F		6	12	24	20	60	120	30	120	360	720	42	210	840	2520	5040
$t = 1$	run	3	4	4	5	5	5	6	6	6	6	7	7	7	7	7
	$D_{\mathbf{l,eff}}$	1.0000	0.8439	0.9641	0.8608	0.9930	0.9968	0.8344	0.8175	0.9881	0.9957	0.8334	0.7369	0.9980	0.9850	0.9958
$t = 2$	run	–	8	8	10	10	10	12	12	12	12	14	14	14	14	14
	$D_{\mathbf{l,eff}}$	–	0.9775	0.9929	1.0000	1.0000	0.9987	0.9753	0.9480	0.9971	0.9985	0.9618	0.9537	0.9995	0.9962	0.9983
	$D_{\mathbf{q,eff}}$	–	–	0.01	0.7947	0.9884	0.7742	0.0493	0.7130	0.7409	0.0488	0.7208	0.7954	0.8452	0.8597	0.6967
$t = 3$	run	–	–	12	15	15	15	18	18	18	18	21	21	21	21	21
	$D_{\mathbf{l,eff}}$	–	–	0.9898	0.9860	0.9992	0.9988	0.9903	0.9815	0.9992	0.9986	0.9859	0.9809	0.9997	0.9979	0.9985
	$D_{\mathbf{q,eff}}$	–	–	0.8773	0.9058	0.9457	0.8410	0.8805	0.9192	0.9587	0.7794	0.8438	0.8975	0.9299	0.9365	0.7744
	$D_{\mathbf{s,eff}}$	–	–	0.0000	–	0.6521	0.0780	–	–	–	–	–	–	–	–	–
$t = 4$	run	–	–	16	–	20	20	24	24	24	24	28	28	28	28	28
	$D_{\mathbf{l,eff}}$	–	–	0.9929	–	1.0000	0.9991	0.9913	0.9769	0.9992	0.9985	0.9878	0.9928	0.9998	0.9979	0.9986
	$D_{\mathbf{q,eff}}$	–	–	0.8309	–	0.9966	0.8327	0.8973	0.9327	0.9563	0.8106	0.8745	0.9534	0.9752	0.9412	0.7701
	$D_{\mathbf{s,eff}}$	–	–	0.0000	–	0.9026	0.7240	–	0.7320	0.7857	0.5943	–	0.7064	0.6517	0.6187	0.1576
$t = 5$	run	–	–	–	–	25	25	–	30	30	30	35	35	35	35	35
	$D_{\mathbf{l,eff}}$	–	–	–	–	0.9997	0.9990	–	0.9818	0.9995	0.9985	0.9942	0.9970	0.9999	0.9985	0.9986
	$D_{\mathbf{q,eff}}$	–	–	–	–	0.9811	0.8231	–	0.9477	0.9648	0.8012	0.8853	0.9578	0.9825	0.9566	0.7754
	$D_{\mathbf{s,eff}}$	–	–	–	–	0.9398	0.6818	–	0.8196	0.8547	0.6367	–	0.8077	0.8052	0.7695	0.5402
$t = 6$	run	–	–	–	–	30	–	–	36	36	36	–	42	42	42	42
	$D_{\mathbf{l,eff}}$	–	–	–	–	1.0000	–	–	0.9834	0.9994	0.9985	–	0.9948	0.9998	0.9984	0.9985
	$D_{\mathbf{q,eff}}$	–	–	–	–	0.9977	–	–	0.9420	0.9741	0.7973	–	0.9607	0.9791	0.9653	0.7811
	$D_{\mathbf{s,eff}}$	–	–	–	–	0.9477	–	–	0.8329	0.8928	0.6054	–	0.8495	0.8411	0.8078	0.5617
$t = 7$	run	–	–	–	–	35	–	–	42	42	–	–	49	49	49	49
	$D_{\mathbf{l,eff}}$	–	–	–	–	0.9999	–	–	0.9831	0.9994	–	–	0.9962	0.9998	0.9985	0.9985
	$D_{\mathbf{q,eff}}$	–	–	–	–	0.9893	–	–	0.9471	0.9741	–	–	0.9656	0.9754	0.9709	0.7715
	$D_{\mathbf{s,eff}}$	–	–	–	–	0.9524	–	–	0.8559	0.8928	–	–	0.8726	0.8638	0.8337	0.5432
$t = 8$	run	–	–	–	–	40	–	–	48	48	–	–	56	56	56	–
	$D_{\mathbf{l,eff}}$	–	–	–	–	0.9998	–	–	0.9836	0.9997	–	–	0.9969	0.9998	0.9985	–
	$D_{\mathbf{q,eff}}$	–	–	–	–	0.9908	–	–	0.9507	0.9797	–	–	0.9687	0.9796	0.9715	–
	$D_{\mathbf{s,eff}}$	–	–	–	–	0.9680	–	–	0.8754	0.9256	–	–	0.8940	0.8883	0.8517	–

Table B2 D-efficiencies for different $m \in \{8, 9\}$, q, and t.

m		8						9						
q		2	3	4	5	6	7	2	3	4	5	6	7	8
run_F		56	336	1680	6720	20160	40320	72	504	3024	15120	60480	181440	362880
$t = 1$	run	8	8	8	8	8	8	9	9	9	9	9	9	9
	$D_{\mathbf{l,eff}}$	0.8258	0.7255	0.7965	0.9733	0.9920	0.9945	0.8234	0.7498	0.7266	0.9826	0.9993	0.9986	0.9989
$t = 2$	run	16	16	16	16	16	16	18	18	18	18	18	18	18
	$D_{\mathbf{l,eff}}$	0.9576	0.9627	0.9498	0.9919	0.9978	0.9983	0.9562	0.9344	0.9462	0.9947	0.9987	0.9996	0.9996
	$D_{\mathbf{q,eff}}$	–	0.7238	0.6332	0.7680	0.7673	0.0988	0.6795	0.7269	0.6657	0.8358	0.7865	0.7992	0.6977
$t = 3$	run	24	24	24	24	24	24	27	27	27	27	27	27	27
	$D_{\mathbf{l,eff}}$	0.9771	0.9786	0.9836	0.9967	0.9990	0.9985	0.9645	0.9729	0.9756	0.9976	0.9994	0.9997	0.9997
	$D_{\mathbf{q,eff}}$	0.8040	0.8813	0.8773	0.9166	0.9001	0.7262	0.7603	0.8438	0.8650	0.8765	0.8944	0.8903	0.7457
$t = 4$	run	32	32	32	32	32	32	36	36	36	36	36	36	36
	$D_{\mathbf{l,eff}}$	0.9820	0.9892	0.9829	0.9983	0.9991	0.9990	0.9740	0.9801	0.9774	0.9980	0.9994	0.9997	0.9997
	$D_{\mathbf{q,eff}}$	0.8523	0.9073	0.9219	0.9428	0.9269	0.7493	0.8177	0.8758	0.9097	0.9220	0.9179	0.9123	0.7572
$t = 5$	run	40	40	40	40	40	40	45	45	45	45	45	45	45
	$D_{\mathbf{l,eff}}$	0.9825	0.9902	0.9836	0.9987	0.9992	0.9988	0.9745	0.9897	0.9829	0.9983	0.9994	0.9997	0.9997
	$D_{\mathbf{q,eff}}$	0.8573	0.9215	0.9387	0.9505	0.9376	0.7570	0.8265	0.9044	0.9279	0.9411	0.9375	0.9163	0.7680
	$D_{\mathbf{s,eff}}$	–	0.7092	0.6630	0.6962	0.6209	0.3939	–	0.4827	0.5247	0.5576	0.5507	0.5294	0.1755
$t = 6$	run	48	48	48	48	48	48	54	54	54	54	54	54	54
	$D_{\mathbf{l,eff}}$	0.9828	0.9925	0.9836	0.9988	0.9992	0.9989	0.9749	0.9900	0.9854	0.9984	0.9995	0.9998	0.9997
	$D_{\mathbf{q,eff}}$	0.8666	0.9277	0.9523	0.9557	0.9448	0.7624	0.8248	0.9015	0.9319	0.9552	0.9560	0.9239	0.7636
	$D_{\mathbf{s,eff}}$	–	0.7357	0.7293	0.7537	0.7098	0.4837	–	0.5785	0.6542	0.6853	0.6934	0.6240	0.4009
$t = 7$	run	–	56	56	56	56	56	63	63	63	63	63	63	63
	$D_{\mathbf{l,eff}}$	–	0.9922	0.9852	0.9990	0.9993	0.9988	0.9755	0.9892	0.9856	0.9983	0.9996	0.9998	0.9997
	$D_{\mathbf{q,eff}}$	–	0.9349	0.9609	0.9652	0.9437	0.7680	0.8324	0.9122	0.9431	0.9658	0.9571	0.9255	0.7649
	$D_{\mathbf{s,eff}}$	–	0.7883	0.7838	0.8142	0.7555	0.5159	–	0.6621	0.7327	0.7505	0.7534	0.6850	0.4453
$t = 8$	run	–	64	64	64	64	64	72	72	72	72	72	72	72
	$D_{\mathbf{l,eff}}$	–	0.9949	0.9856	0.9990	0.9992	0.9989	0.9746	0.9922	0.9877	0.9986	0.9995	0.9998	0.9997
	$D_{\mathbf{q,eff}}$	–	0.9414	0.9538	0.9646	0.9472	0.7641	0.8363	0.9246	0.9485	0.9629	0.9579	0.9308	0.7647
	$D_{\mathbf{s,eff}}$	–	0.8225	0.8059	0.8494	0.7636	0.5038	–	0.7164	0.7588	0.7899	0.7849	0.7042	0.4669
$t = 9$	run	–	72	72	72	72	–	–	81	81	81	81	81	81
	$D_{\mathbf{l,eff}}$	–	0.9935	0.9874	0.9992	0.9993	–	–	0.9934	0.9896	0.9987	0.9996	0.9998	0.9997
	$D_{\mathbf{q,eff}}$	–	0.9400	0.9601	0.9680	0.9489	–	–	0.9215	0.9561	0.9671	0.9638	0.9340	0.7605
	$D_{\mathbf{s,eff}}$	–	0.8275	0.8314	0.8679	0.7834	–	–	0.7138	0.7882	0.8268	0.8179	0.7277	0.4538
$t = 10$	run	–	80	80	80	80	–	–	90	90	90	90	90	–
	$D_{\mathbf{l,eff}}$	–	0.9962	0.9857	0.9994	0.9993	–	–	0.9931	0.9903	0.9987	0.9996	0.9998	–
	$D_{\mathbf{q,eff}}$	–	0.9440	0.9601	0.9722	0.9498	–	–	0.9271	0.9564	0.9696	0.9657	0.9388	–
	$D_{\mathbf{s,eff}}$	–	0.8343	0.8484	0.8769	0.7926	–	–	0.7567	0.8195	0.8310	0.8381	0.7334	–
$t = 11$	run	–	88	88	88	88	–	–	99	99	99	99	99	–
	$D_{\mathbf{l,eff}}$	–	0.9957	0.9858	0.9994	0.9993	–	–	0.9937	0.9912	0.9987	0.9996	0.9998	–
	$D_{\mathbf{q,eff}}$	–	0.9442	0.9631	0.9767	0.9507	–	–	0.9239	0.9584	0.9731	0.9715	0.9347	–
	$D_{\mathbf{s,eff}}$	–	0.8435	0.8516	0.8952	0.8046	–	–	0.7660	0.8247	0.8528	0.8522	0.7466	–

Table B3 D-efficiencies for $m = 10$, and different q, and t.

m		10							
q		2	3	4	5	6	7	8	9
run_F		90	720	5040	30240	151200	604800	1814400	3628800
$t = 1$	run	10	10	10	10	10	10	10	10
	$D_{\mathrm{l,eff}}$	0.8201	0.6963	0.7251	0.7799	0.9294	0.9852	0.9966	0.9982
$t = 2$	run	20	20	20	20	20	20	20	20
	$D_{\mathrm{l,eff}}$	0.9404	0.9445	0.9468	0.9473	0.9979	0.9987	0.9983	0.9993
	$D_{\mathrm{q,eff}}$	–	0.6665	0.6152	0.6646	0.7396	0.7434	0.9267	0.1527
$t = 3$	run	30	30	30	30	30	30	30	30
	$D_{\mathrm{l,eff}}$	0.9552	0.9689	0.9781	0.9703	0.9922	0.9976	0.9992	0.9994
	$D_{\mathrm{q,eff}}$	0.7192	0.8159	0.8311	0.8451	0.8966	0.8830	0.8640	0.7017
$t = 4$	run	40	40	40	40	40	40	40	40
	$D_{\mathrm{l,eff}}$	0.9657	0.9766	0.9794	0.9820	0.9932	0.9982	0.9994	0.9992
	$D_{\mathrm{q,eff}}$	0.7697	0.8615	0.8762	0.8968	0.9178	0.9117	0.8825	0.7268
$t = 5$	run	50	50	50	50	50	50	50	50
	$D_{\mathrm{l,eff}}$	0.9644	0.9813	0.9831	0.9813	0.9943	0.9983	0.9992	0.9983
	$D_{\mathrm{q,eff}}$	0.7994	0.8760	0.9054	0.9232	0.9329	0.9246	0.8949	0.7248
	$D_{\mathrm{s,eff}}$	–	–	–	–	–	–	–	–
$t = 6$	run	60	60	60	60	60	60	60	60
	$D_{\mathrm{l,eff}}$	0.9658	0.9870	0.9849	0.9825	0.9940	0.9983	0.9985	0.9985
	$D_{\mathrm{q,eff}}$	0.7977	0.8879	0.9191	0.9318	0.9406	0.9395	0.8990	0.7491
	$D_{\mathrm{s,eff}}$	–	0.5032	0.5886	0.5966	0	0	0	0
$t = 7$	run	70	70	70	70	70	70	70	70
	$D_{\mathrm{l,eff}}$	0.9674	0.9868	0.9879	0.9854	0.9954	0.9986	0.9995	0.9995
	$D_{\mathrm{q,eff}}$	0.8025	0.8909	0.9417	0.9428	0.9497	0.9453	0.9092	0.7302
	$D_{\mathrm{s,eff}}$	–	0.5815	0.6690	0.6953	0	0	0	0
$t = 8$	run	80	80	80	80	80	80	80	80
	$D_{\mathrm{l,eff}}$	0.9674	0.9891	0.9909	0.9834	0.9957	0.9988	0.9993	0.9980
	$D_{\mathrm{q,eff}}$	0.8059	0.8983	0.9383	0.9452	0.9537	0.9530	0.9069	0.7438
	$D_{\mathrm{s,eff}}$	–	0.6289	0.7239	0.7283	0	0	0	0
$t = 9$	run	–	90	90	90	90	90	90	90
	$D_{\mathrm{l,eff}}$	–	0.9897	0.9911	0.9845	0.9963	0.9988	0.9983	0.9992
	$D_{\mathrm{q,eff}}$	–	0.9055	0.9419	0.9490	0.9647	0.9510	0.9076	0.7316
	$D_{\mathrm{s,eff}}$	–	0.6573	0.7542	0.7769	0	0	0	0
$t = 10$	run	–	100	100	100	100	100	100	100
	$D_{\mathrm{l,eff}}$	–	0.9920	0.9915	0.9845	0.9960	0.9985	0.9979	0.9984
	$D_{\mathrm{q,eff}}$	–	0.9068	0.9464	0.9570	0.9643	0.9566	0.9068	0.7367
	$D_{\mathrm{s,eff}}$	–	0.6694	0.7762	0.7982	0	0	0	0
$t = 11$	run	–	110	110	110	110	110	110	–
	$D_{\mathrm{l,eff}}$	–	0.9905	0.9923	0.9837	0.9967	0.9986	0.9993	–
	$D_{\mathrm{q,eff}}$	–	0.9070	0.9480	0.9557	0.9751	0.9595	0.9084	–
	$D_{\mathrm{s,eff}}$	–	0.6858	0.7993	0.8102	0	0	0	–