Factorial Designs under Baseline Parameterization and Space-filling Designs with Applications to Big Data

by

Cheng-Yu Sun

M.Sc., National Tsing Hua University, 2014
B.Sc., National Tsing Hua University, 2012

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

in the
Department of Statistics and Actuarial Science
Faculty of Science

SIMON FRASER UNIVERSITY
Summer 2021

Declaration of Committee

Name:	**Cheng-Yu Sun**
Degree:	**Doctor of Philosophy**
Thesis title:	**Factorial Designs under Baseline Parameterization and Space-filling Designs with Applications to Big Data**
Committee:	**Chair:** Joan Hu Professor, Statistics and Actuarial Science
	Boxin Tang Supervisor Professor, Statistics and Actuarial Science
	Richard Lockhart Commitee Member Professor, Statistics and Actuarial Science
	Liangliang Wang Examiner Associate Professor, Statistics and Actuarial Science
	J.P. Morgan External Examiner Professor, Statistics Virginia Tech

Abstract

This dissertation reports my research work on three topics in the areas of two-level factorial designs under the baseline parameterization, space-filling designs, and sub-data selection for big data. When studying two-level factorial designs, factorial effects are usually given by the orthogonal parameterization. But if each factor has an intrinsic baseline level, the baseline parameterization is a more appropriate alternative. We obtain a relationship between these two types of parameterization, and show that certain design properties are invariant. The relationship also allows us to construct an attractive class of robust baseline designs. We then consider two classes of space-filling designs driven by very different considerations: uniform projection designs and strong orthogonal arrays (SOAs), where the former are obtained by minimizing the uniform projection criterion while the latter are a special kind of orthogonal arrays. We express the uniform projection criterion in terms of the stratification characteristics related to an SOA. This new expression is then used to show that certain SOAs are optimal or nearly optimal under the uniform projection criterion. Finally, we consider the problem of selecting a representative sub-dataset from a big dataset for the purpose of statistical analyses without massive computation. Under the nonparametric regression situation, we present a two-phase selection method, which embodies two important ideas. First, the sub-dataset should be a space-filling subset within the full dataset. Second, in the area where the response surface is more rugged, more data points should be selected. Simulations are conducted to demonstrate the usefulness of our method.

Keywords: centered L_2-discrepancy; computer experiment; minimum aberration, orthogonal array; sub-data selection

To my beloved wife and parents,

and to my daughter, Mu-Yen

Acknowledgements

I would first like to offer my sincere thanks to my senior supervisor, Professor Boxin Tang. You brought me to a different level in doing research. I learned from you how to see the big picture and how to stay in a right direction when investigating research problems. I admire you that you can explain profound theories in lucid and concise ways. You are definitely my role model as a researcher. I feel so lucky to be your student!

I extend my gratitude to the committee members for spending their precious time reading my thesis and making valuable comments. Thank you to Professor Liangliang Wang for serving as the internal examiner, and for your help on my job hunting. Thank you to Professor Joan Hu for chairing my committee. Thank you to Professor Richard Lockhart for serving as the supervisor. Sincere thanks go to my external examiner Professor J.P. Morgan from Virginia Tech, for your detailed and constructive suggestions on my thesis, especially for Chapter 2.

Thank you to Professor Shao-Wei Cheng for bringing me to the research area of experimental designs, and for your support these years. Thank you to Dr. Frederick Kin Hing Phoa for guiding me in doing research before I started my PhD study.

I would like to thank all the department faculty and staff members. It is really nice to have such a friendly environment during my PhD study. Many thanks go to Charlene, Kelly, Sadika, and Jay for always being their when I need your help. I would also like to thank my fellow students, especially Chenlu Shi, Anqi Chen, Wenlong Li, Guanzhou Chen, Yuping Yang, Charlie Zhou, Zhiyang Zhou, and Trevor Thompson. Thank you for your company at SFU, and for the interesting discussions we have had.

Last but not least, I am deeply grateful to my family for their unconditional love and encouragement. Most of all, I would like to express my special thanks to my wife, Peggie, who brought me the best gift of my life, my beloved daughter, Mu-Yen.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Statistical data analysis and model building play a vital role in almost every field of scientific studies nowadays. Generally speaking, a statistical model contains a dependent variable called the response, and several independent variables called covariates or explanatory variables. Through probability theory, a statistical model gives a relationship between the dependent and independent variables. Statisticians use available data to draw inferences related to the model, and then provide answers to the research questions. But data collection may be costly, either in terms of time or money, so how to collect the data economically is critically important in statistics. If the explanatory variables are subject to the researchers' control, we have an opportunity of designing a study. For different research purposes, there are different types of designs. This thesis presents my research in three areas of experimental design. Chapter 2 considers two-level factorial designs under the baseline parameterization, Chapter 3 examines space-filling designs for computer experiments, and Chapter 4 utilizes the idea of space-filling designs to investigate the sub-data selection problem for big data. The following paragraphs provide the synopses of these chapters.

Chapter 2 considers the experiments that involve m factors, each of two levels. Since there are 2^m possible treatment means, there are 2^m parameters in a full model. Generally, a full model is hardly ever considered because it violates the principle of Occam's razor. The most often built models are those that contain only a few important parameters representing the change of the mean response caused by the level changing of factors, called factorial effects. The definition of factorial effects is commonly given by the orthogonal parameterization (OP). The baseline parameterization (BP) is a less often used but more appropriate alternative in some experimental situations. For example, Yang and Speed [1], Kerr [2], and Banerjee and Mukerjee [3] considered the cDNA microarray experiments under the BP. In experimental design literature, most work on two-level factorial designs directly adopts the OP without explicitly defining the factorial effects, but many results may not apply to the BP. For example, Mukerjee and Tang [4] investigated minimum aberration designs under the BP and found that the optimal designs are not the same as, though related to, those under the OP. We study the relationship between these two types of parameterization, and examine its applications to design construction. We first show that the factorial effects under the BP depend on those under the OP in a special way, and vice versa. Then we establish the equivalence of echelon models under the two parameterizations, which gives rise to the invariance of some design properties, including the estimability and the D-efficiency. Moreover, we generalize the optimality property of an orthogonal design given by Moriguti [5]. The results of Chapter 2 have been published in Sun and Tang [6], and a further applied work has been published in Chen, Sun, and Tang [7].

Computer modeling is routinely used in modern scientific investigations. In a computer experiment, the true model is represented by a computer code, so complex that producing an output is very time-consuming. Thus, it is often required to build a surrogate model that is easy to compute, based on the data consisting of the outputs under a chosen set of input settings. Space-filling designs are suitable for the purpose of selecting input settings. The basic idea is to spread design points throughout the experimental region in an uniform fashion. Such a design can be obtained by optimizing a distance or discrepancy criterion. Popular criteria include the maximin distance criterion [8] and the centered L_2-discrepancy [9]. Optimal designs of this kind may not enjoy uniformity properties when projected into low dimensions, which is important because among a large number of factors, usually only a few of them are active. Designs with good projection properties are desirable for factor screening [see 10, 11]. Orthogonal arrays are a class of designs that are space-filling in low dimensions. For example, Latin hypercube designs [12] are orthogonal arrays of strength one. In Chapter 3, we consider uniform projection designs and strong orthogonal arrays (SOAs) of strength 2+, two classes of space-filling designs that both focus on two-dimensional projection properties but are motivated by different considerations. The former, introduced by Sun, Wang, and Xu [13], are obtained by minimizing the uniform projection criterion, while the latter, studied by He, Cheng, and Tang [14], are a special kind of orthogonal arrays. We find that the uniform projection criterion can be written as a weighted sum of squares, where each square measures one aspect of uniformity of design points. This new expression provides insights into the uniform projection criterion, and can be used to show that an SOA of strength 2+ is optimal or nearly optimal under the uniform projection criterion. The results of Chapter 3 have been written into a paper and submitted to the Journal of the American Statistical Association, which is now under the second round review after a moderate revision.

Chapter 4 considers the problem of sub-data selection for big data. Conducting statistical analysis on a big dataset is challenging, as no matter what kind of model one uses, the volume of the dataset makes model fitting extremely difficult. One solution is to find a representative subset of the dataset and fit a model using the sub-dataset. For example, Ma, Mahoney, and Yu [15] discussed a sampling method under the linear regression, in which the sampling probabilities are based on the normalized statistical leverage scores. More recently, Wang, Yang, and Stufken [16] presented a deterministic method that selects the extreme data points in order to have a good estimation under the linear regression. Instead of a linear model, we consider in Chapter 4 nonparametric estimation of the unknown mean response. Utilizing the idea of space-filling designs, we develop an algorithm that selects an orthogonal array-structured sub-dataset. Based on this algorithm, we then propose a two-phase sub-data selection method, which allows more data points to be selected in the areas where the response surface is more rugged. Simulations are carried out to illustrate the usefulness of our method. A salient feature of our method is that it makes use of the available information on the response for the first time in the literature.

In each of Chapters 2-4, the main contents start with an introduction and end by a concluding section. Chapters 2 and 3 have proofs in separate sections. To avoid complex notation, we only make effort to have consistent notation within each chapter. A summary of this thesis is given in Chapter 5, where some possible future research is also discussed.

Chapter 2

Relationship between orthogonal and baseline parameterizations and its applications to design constructions

2.1 Introduction

In many industrial and scientific investigations, the objective is to build a model that can adequately describe how the response of a system changes when the levels of the input factors change. The impact on the mean response caused by changing the levels of one or more factors is called a factorial effect. The most commonly adopted definition of factorial effects for a 2^m factorial, given by Box and Hunter [17], is a set of mutually orthogonal treatment contrasts, called the orthogonal parameterization (OP). Despite having received less attention, a more appropriate alternative in some situations is the baseline parameterization (BP). Under the BP, experimenters are more interested in the effects when non-involved factors are kept at their intrinsic baseline levels.

The BP is relatively underexplored, but is becoming more important. Yang and Speed [1], Kerr [2], and Banerjee and Mukerjee [3] investigated factorial designs under the BP in the context of cDNA microarray experiments. More recently, Mukerjee and Tang [4] proposed a minimum K-aberration criterion to sequentially minimize the bias in the estimation of main effects caused by non-negligible interactions, in the order of importance given by the effect hierarchical principle (Wu and Hamada [18], pp.172–3). The construction of minimum

K-aberration designs is further considered in Li, Miller, and Tang [19], Miller and Tang [20], and Mukerjee and Tang [21].

Because the factorial effects under the OP and BP are both treatment contrasts, there must exist a linear relationship between them. What cannot be foreseen is the special way one set of effects depends on the other. This special pattern in the linear relationship has some important implications in the construction of baseline designs. We aim to derive this relationship and explore its applications to design construction under the BP in terms of estimability, optimality, and robustness.

The rest of this chapter is organized as follows. In Section 2.2, we first provide formal definitions of factorial effects under the OP and the BP. Then, we derive the linear relationship between the two types of parameterization and examine its implications. Section 2.3 shows how to use the results in Section 2.2 to find designs under the BP. Here we show that certain orthogonal arrays continue to be optimal under the BP. General Rechtschaffner designs are introduced, and are shown to enjoy a robust property under the BP. A numerical study is given in Section 2.4, and Section 2.5 concludes the chapter. All proofs are given in Section 2.6.

2.2 Relationship between the OP and BP

Consider a factorial experiment involving m two-level factors $F_1, F_2, \dots, F_m$, each at levels zero and one. Let τ_g denote the mean response at the treatment combination $g = (g_1, g_2, \dots, g_m)$, with $g_i = 0$ or 1 $(i = 1, 2, \dots, m)$, and let $\mathcal{G}$ be the collection of all 2^m treatment combinations. Because the treatment combination $(1, 1, 0, \dots, 0)$ corresponds to the subset $\{1, 2\}$ of $S = \{1, 2, \dots, m\}$, we use τ_{12} and $\tau_{(1,1,0,\dots,0)}$ interchangeably, depending on which one is more convenient within the context. Under the OP, for a subset $v = \{i_1, i_2, \cdots, i_k\}$ of S, the k-factor interaction $F_{i_1} F_{i_2} \cdots F_{i_k}$ (the main effect if $k = 1$) is given by

$$\beta_v = \frac{1}{2^m} \sum_{g \in \mathcal{G}} \tau_g (-1)^{\sum_{h=1}^{k} g_{i_h}}. \tag{2.1}$$

We let $\beta_\phi = 2^{-m} \sum_{\mathcal{G}} \tau_g$, which is the grand mean. Under the BP, the main effect of F_i is given by $\theta_i = \tau_i - \tau_\phi$, and the two-factor interaction $F_i F_j$ is given by $\theta_{ij} = \tau_{ij} - \tau_i - \tau_j + \tau_\phi$.

More generally, for a subset $w = \{i_1, i_2, \cdots, i_k\}$ of S, the k-factor interaction $F_{i_1}F_{i_2}\cdots F_{i_k}$ under the BP is given by

$$\theta_w = \sum_{u \subseteq w} \tau_u (-1)^{|w|-|u|}, \tag{2.2}$$

where $|\cdot|$ stands for the cardinality of a set.

Both β_v and θ_w measure the impact on τ_g caused by level changing of the involved factor(s). However, the former considers an overall effect, whereas the latter focuses on the situation in which all non-involved factors are set at level zero, the baseline level. For example, consider $v = w = \{1\}$ in (2.1) and (2.2). Let $\mathcal{G}^* = \{(g_2, g_3, \ldots, g_m) : g_i = 0, 1\}$. The main effects of F_1 under the OP and the BP can be written as $\beta_1 = (1/2^m)\sum_{g^* \in \mathcal{G}^*}\left(\tau_{(0,g^*)} - \tau_{(1,g^*)}\right)$ and $\theta_1 = \tau_{(1,0,\ldots,0)} - \tau_{(0,0,\ldots,0)}$, respectively. Up to a constant, β_1 averages out the effects of F_1 conditional on every $g^* \in \mathcal{G}^*$, while θ_1 computes only the effect of F_1 when all other factors are set at their baseline levels.

The BP arises naturally when each factor has a null state or a baseline level. For example, in a toxicological study, each factor is a toxin, and each treatment combination is a mix of several toxins. The absence and presence of a particular toxin can be represented by levels zero and one, respectively. In an agricultural experiment, two kinds of fertilizers may be applicable, serving as the two levels of a factor. Then level zero can stand for the currently used fertilizer, and level one for the new fertilizer.

By combining (2.1) and (2.2), we obtain a linear relationship between the OP and BP, as stated in the following theorem.

Theorem 2.1. *We have that*

(i) $\beta_v = \sum_{w \supseteq v} a_w \theta_w$, *with* $a_w = (-1)^{|v|}2^{-|w|}$,

(ii) $\theta_w = \sum_{v \supseteq w} c_v \beta_v$, *with* $c_v = (-2)^{|w|}$.

In Theorem 2.1, the θ_w's in the expression of β_v are those with w containing v. A similar phenomenon occurs in the expression of θ_w in terms of β_v. It is this special pattern in the linear relationship between θ_w and β_v that makes it useful in the construction of baseline designs, which we examine in Section 2.3. Proposition 2 in Mukerjee and Tang [4], which states that an orthogonal array is universally optimal for estimating the main effects under the BP, is established based on the simple fact that $\theta_i = -2\beta_i$, for $i = 1, 2, \ldots, m$, if $\beta_v = 0$ for all $|v| \geq 2$. A more important implication is that the absence of interactions under the OP yields the same result under the BP, and vice versa. We now consider a situation that is more general than the absence of interactions. For a collection $\mathcal{C}$ of subsets of S, we say it is *echelon* if for any s collected by $\mathcal{C}$, all subsets of s are also collected. Then, Theorem 2.1 implies the following result.

Corollary 2.1. *Let $\mathcal{C}$ be echelon. Then, $\beta_v = 0$ for all $v \notin \mathcal{C}$, if and only if $\theta_w = 0$ for all $w \notin \mathcal{C}$. As a special case, the absence of factorial effects of order k or higher is invariant to the choice of the parameterization.*

If a collection of factorial effects, say $\{\beta_v : v \in \mathcal{C}\}$ or $\{\theta_w : w \in \mathcal{C}\}$, are believed to be active, the corresponding models under the OP and BP are, respectively,

$$\tau_g = \sum_{v \in \mathcal{C}} \beta_v \prod_{k \in v} (1 - 2g_k) \ (g \in \mathcal{G}); \tag{2.3}$$

$$\tau_g = \sum_{w \in \mathcal{C}} \theta_w \prod_{k \in w} g_k \ (g \in \mathcal{G}). \tag{2.4}$$

We say that models (2.3) and (2.4) are, respectively, the OP and the BP models associated with $\mathcal{C}$, and are called echelon if $\mathcal{C}$ is echelon. Corollary 2.1 states that these two models are equivalent if $\mathcal{C}$ is echelon. The main-effect-only model and the models that contain all of the main effects, plus some/all of the two-factor interactions, are most often used in practice, all of which are echelon models. We end this section with two toy examples that illustrate Theorem 2.1 and Corollary 2.1.

Example 2.1. *Consider a three-factor system A, with mean responses given by*

$$\textit{System A:}\ (\tau_{000}, \tau_{001}, \tau_{010}, \tau_{011}, \tau_{100}, \tau_{101}, \tau_{110}, \tau_{111}) = (1, 1, 1, 1, 2, 2, 5, 5).$$

By equation (2.2), there are only two active factorial effects under the BP: $\theta_1 = 1$ and $\theta_{12} = 3$. However, by equation (2.1), there are three active factorial effects under the OP: $\beta_1 = -1.25$, $\beta_2 = -0.75$, and $\beta_{12} = 0.75$. The OP model that contains only β_1 and β_{12} fails to characterize the mean response structure, because $\mathcal{C} = \{\phi, \{1\}, \{1, 2\}\}$ is not an echelon *collection. Applying part (i) of Theorem 2.1,* $\beta_{12} = 0.25\theta_{12} + 0.125\theta_{123} = 0.75$. *One can compute* β_v *similarly for other* v.

Example 2.2. *A second system has the following mean responses:*

$$\textit{System B: } (\tau_{000}, \tau_{001}, \tau_{010}, \tau_{011}, \tau_{100}, \tau_{101}, \tau_{110}, \tau_{111}) = (1, 1, -1, -1, 2, 2, 3, 3).$$

Under the BP, $(\theta_1, \theta_2, \theta_{12}) = (1, -2, 3)$, *and all other* θ_w *are zero. Because the model is associated with an echelon collection* $\mathcal{C} = \{\phi, \{1\}, \{2\}, \{1, 2\}\}$, *by Corollary 2.1, the OP model that contains only* β_1, β_2, *and* β_{12} *is true as well. Using equation (2.1) to verify this, we find that* $(\beta_1, \beta_2, \beta_{12}) = (-1.25, 0.25, 0.75)$, *and all other* β_v *are zero.*

2.3 Finding baseline designs

2.3.1 Preliminary results

Suppose N experimental runs are allowed in a design $\mathcal{D}$, and let $(g_{i1}, g_{i2}, \cdots, g_{im})$ denote the ith run $(i = 1, 2, \ldots, N)$. Under design $\mathcal{D}$, the OP and BP models associated with $\mathcal{C}$ are, respectively,

$$E(Y_i) = \sum_{v \in \mathcal{C}} \beta_v \prod_{j \in v} (1 - 2g_{ij}) \ (i = 1, 2, \ldots, N); \tag{2.5}$$

$$E(Y_i) = \sum_{w \in \mathcal{C}} \theta_w \prod_{j \in w} g_{ij}, \ (i = 1, 2, \ldots, N), \tag{2.6}$$

where Y_i is the response of the ith run. Let $X_{\mathcal{C}}$ and $W_{\mathcal{C}}$ be the model matrices of (2.5) and (2.6), respectively. A design is said to be able to estimate model (2.5) (respectively, model (2.6)) if $X'_{\mathcal{C}}X_{\mathcal{C}}$ (respectively, $W'_{\mathcal{C}}W_{\mathcal{C}}$) is invertible.

Theorem 2.2. *If a design is able to estimate an echelon OP model, it is able to estimate its counterpart BP model, and vice versa.*

Theorem 2.2 allows the estimability of certain BP models to be established with little effort. One example is that the full kth-order model, the model that contains all factorial effects of order k or lower, can be estimated under an orthogonal array of strength $2k$. Another interesting application of Theorem 2.2 is given in the next example.

Example 2.3. *Cheng [22] showed that an N-run orthogonal array, if N is not a multiple of eight, can estimate the full second-order model when projected onto any four factors. This projection property, by Theorem 2.2, holds regardless of the parameterization.*

For a design $\mathcal{D}$ and an OP model associated with $\mathcal{C}$, we define its $D_{\mathcal{C}}$-efficiency as $\det(X_{\mathcal{C}}'X_{\mathcal{C}})$, and its $A_{\mathcal{C}}$-efficiency as $\operatorname{trace}(X_{\mathcal{C}}'X_{\mathcal{C}})^{-1}$. We say a design is $D_{\mathcal{C}}$-optimal (respectively, $A_{\mathcal{C}}$-optimal) if it maximizes $\det(X_{\mathcal{C}}'X_{\mathcal{C}})$ (respectively, minimizes $\operatorname{trace}(X_{\mathcal{C}}'X_{\mathcal{C}})^{-1}$) among all competing designs. Similarly, we can define the $D_{\mathcal{C}}$- and $A_{\mathcal{C}}$-optimality criteria under the BP by replacing $X_{\mathcal{C}}$ with $W_{\mathcal{C}}$.

Proposition 2.1. *Let $\mathcal{C}$ be an echelon collection. If a design is $D_{\mathcal{C}}$-optimal under the OP, it is $D_{\mathcal{C}}$-optimal under the BP, and vice versa.*

Proposition 2.1 is an implication of a more general result given by Proposition 2.2, which can be derived directly from Theorem 3.1. Note that Propositions 2.1 and 2.2 are both special cases of Lemma 6 in Stallings and Morgan [23], though stated in a different context.

Proposition 2.2. *If $\mathcal{C}$ is echelon, then $\det(X_{\mathcal{C}}'X_{\mathcal{C}})$ is proportional to $\det(W_{\mathcal{C}}'W_{\mathcal{C}})$. The ratio does not depend on the design, but on $\mathcal{C}$ alone.*

We conclude this subsection with a corollary. Its implication will be discussed after Theorem 3 in the next subsection.

Corollary 2.2. *Let $\mathcal{C}$ be an echelon collection. The $D_{\mathcal{C}}$-efficiency of a design remains unchanged under level switching of one or more factors, regardless of the parameterization.*

2.3.2 Designs from orthogonal arrays

Cheng [24] showed that an orthogonal array is universally optimal under the main-effect-only model. As another example, a design given by an orthogonal array of strength $2k$ is A- and D-optimal under the full kth-order model. These results are all obtained all under the OP. In this subsection, we generalize a result of Moriguti [5] to baseline designs. We also comment on generating baseline designs with robust properties.

Consider the OP model associated with $\mathcal{C}$, and let $\hat{\beta}_v$ be the least squares estimator of β_v. We assume, as usual, that all observations are uncorrelated and have a common variance.

Moriguti [5] proved that a design in which the model matrix $X_{\mathcal{C}}$ has mutually orthogonal columns minimizes $\text{Var}(\hat{\beta}_v)$ for each $v \in \mathcal{C}$ among all competing designs. The next theorem states that a similar result holds for the BP if $\mathcal{C}$ is echelon.

Theorem 2.3. *Under an OP model associated with $\mathcal{C}$, a design $\mathcal{D}$ minimizes $Var(\hat{\beta}_v)$ for each $v \in \mathcal{C}$ among all competing designs if $X_{\mathcal{C}}$ is orthogonal. Furthermore, if $\mathcal{C}$ is echelon, then under the counterpart BP model, $\mathcal{D}$ also minimizes $Var(\hat{\theta}_w)$ among all competing designs for every w in $\mathcal{C}$ that is not contained by another u in $\mathcal{C}$.*

For convenience, we call θ_w a *cap effect* if w is not contained by another u in $\mathcal{C}$. Then, Theorem 3 establishes the optimality for every cap effect under the stated conditions. Cap effects should be tested first for their significance when seeking a simpler model in the analysis stage. We consider some useful cases. If the main-effects model is considered with the inclusion of an intercept, then all the main effects are cap effects. Therefore, Theorem 3 generalizes a result of Mukerjee and Tang [4], who established the optimality for every main effect. For a model consisting of all main effects and all two-factor interactions, the two-factor interactions are cap effects. In a model of all main effects plus some two-factor interactions, these two-factor interactions are cap effects, as are the main effects not involved in these two-factor interactions.

Because switching the two levels does not affect the orthogonality of $X_{\mathcal{C}}$, Theorem 3 also suggests a simple strategy for generating an efficient baseline design that is robust to non-negligible effects. While a full investigation of this problem is beyond the scope of this study, we give an example to illustrate the idea.

Example 2.4. *Consider the model associated with $\mathcal{C} = \{\phi, \{1\}, \{2\}, \{3\}, \{4\}, \{1,2\}, \{1,3\}\}$ and an eight-run design $\mathcal{D}$, displayed in transposed form below:*

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Design $\mathcal{D}$ is a resolution-IV regular design. Because the design has an orthogonal model matrix $X_{\mathcal{C}}$, it has the optimal properties given in Theorem 3. Let $\mathcal{D}^*$ be the design obtained *from $\mathcal{D}$ by level switching the fourth factor. Then, $\mathcal{D}^*$ has the same optimality properties as $\mathcal{D}$. To further distinguish one design from the other, we compute the bias caused by non-negligible effects. Assume θ_{24} is the only non-negligible effect. Following the idea of the minimum K-aberration, the design with smaller value of $\|(W_{\mathcal{C}}'W_{\mathcal{C}})^{-1}W_{\mathcal{C}}'W_{24}\|$ is preferred, where $W_{\mathcal{C}}$ is the model matrix under the BP, W_{24} is the Hadarmard product of the second and fourth factors in the design matrix, and $\|\cdot\|$ denotes the Euclidean norm. Because $\|(W_{\mathcal{C}}'W_{\mathcal{C}})^{-1}W_{\mathcal{C}}'W_{24}\|$ is equal to 2 for $\mathcal{D}$ and 0.816 for $\mathcal{D}^*$, $\mathcal{D}^*$ is preferred.*

2.3.3 Rechtschaffner designs

Consider the full second-order model associated with the collection $\mathcal{C}_2 = \{s \subseteq S : |s| \leq 2\}$. Based on the aforementioned one-to-one correspondence between a subset and a treatment combination, $\mathcal{C}_2$ corresponds to a design consisting of $(1 + m + m(m-1)/2)$ different treatment combinations, which is known as the Rechtschaffner design, denoted by $\mathcal{D}_{\mathcal{C}_2}$. Using the same correspondence, we define $\mathcal{D}_{\mathcal{C}}$ similarly for any $\mathcal{C}$, and still call it a Rechtschaffner design. Design $\mathcal{D}_{\mathcal{C}_2}$ was first presented by Rechtschaffner [25], who suggested its use under the full second-order model. The estimability of $\mathcal{D}_{\mathcal{C}_2}$ under the OP was later proved by several authors, with generalizations to echelon models for mixed-level and/or higher-order situations. We state a result for the two-level situation, which is a special case of Theorem 15.25 in Cheng [26].

Proposition 2.3. *For an echelon collection $\mathcal{C}$, the OP model associated with $\mathcal{C}$ is estimable under the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$.*

Under the BP, the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$ has a stronger property.

Theorem 2.4. *For any collection $\mathcal{C}$, the BP model associated with $\mathcal{C}$ is estimable under the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$.*

Compared with Proposition 2.3, Theorem 2.4 does not assume that $\mathcal{C}$ is echelon. A special case of Rechtschaffner designs is $\mathcal{D}_{\mathcal{C}_1}$ with $\mathcal{C}_1 = \{s \subseteq S : |s| \leq 1\}$. This design, commonly known as a one-factor-at-a-time design, was discussed in Mukerjee and Tang [4] for its following robust property: non-negligible interactions never cause bias in the estimation of the main effects under the BP. This property, in fact, holds for any Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$ with an echelon $\mathcal{C}$.

Theorem 2.5. *Let $\mathcal{C}$ be an echelon collection. Then, the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$ allows an unbiased estimation of the BP model associated with $\mathcal{C}$, even if the effects outside the model are non-negligible.*

Example 2.5. *Consider the model associated with $\mathcal{C} = \{\phi, \{1\}, \{2\}, \{3\}, \{4\}, \{1,2\}, \{1,3\}\}$ and the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$, displayed in transposed form below:*

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

If θ_{24} is a non-negligible effect, the bias it causes can be found using $(W'_{\mathcal{C}}W_{\mathcal{C}})^{-1}W'_{\mathcal{C}}W_{24}\theta_{24}$. It is clear that W_{24} is an all-zero vector; hence, θ_{24} does not cause bias in $\hat{\theta}_w$, for all $w \in \mathcal{C}$. The same argument can be made for all other effects outside the model.

Though the Rechtschaffner design $\mathcal{D}_{\mathcal{C}}$ enjoys a nice property of robustness, it is not very efficient. We now consider a class of N-run Rechtschaffner designs based on $\mathcal{D}_{\mathcal{C}}$, where $\mathcal{C} = \{s_0 = \phi, s_1, s_2, \ldots, s_p\}$, by allowing each run in $\mathcal{D}_{\mathcal{C}}$ to appear multiple times. Let f_j be the number of times the treatment combination corresponding to s_j appears in $\mathcal{D}_{\mathcal{C}}$, for $j = 0, 1, \ldots, p$, where $N = \sum_{j=0}^{p} f_j$. The next result gives an optimal allocation.

Proposition 2.4. *Let $\mathcal{C}$ be an echelon collection. An N-run Rechtschaffner design based on $\mathcal{D}_{\mathcal{C}}$ is $A_{\mathcal{C}}$-optimal under the BP if $f_j = Nq_j^{1/2}/\sum_{j=0}^{p} q_j^{1/2}$, for $j = 0, 1, \ldots, p$, where q_j is the number of subsets in $\mathcal{C}$ that contain s_j.*

When the f_j's are not all integers, which is usually the case, one can simply round up or down $f_j = Nq_j^{1/2}/\sum_{j=0}^{p} q_j^{1/2}$ under the constraint $\sum_j f_j = N$. In fact, Proposition 2.4 is a special case of Theorem 2 in Stallings and Morgan [23], where the exact $A_{\mathcal{C}}$-optimal designs among all competing designs are also derived. Their result is more useful since the optimality given by Proposition 2.4 is only within the class of N-run Rechtschaffner designs.

2.4 A numerical study

In this section, we carry out a numerical study to examine the efficiencies of the baseline designs given by Theorem 2.3 and Proposition 2.4 among all competing designs. This numerical study can help us answer two questions. First, compared with the most efficient designs, how efficient is a Rechtschaffner design? Second, when a design given by Theorem 2.3 is used for its maximal efficiency on the estimation of cap effects, how much is sacrificed on the estimation of non-cap effects?

Consider a two-level factorial involving three factors. Suppose sixteen experimental runs are allowed. As usual, we assume all observations are uncorrelated and have a common variance σ^2. Let $f_0, f_1, f_2, f_3, f_4, f_5, f_6$, and f_7 be the numbers of times the level combinations $(0,0,0), (0,0,1), (0,1,0), (1,0,0), (0,1,1), (1,0,1), (1,1,0)$, and $(1,1,1)$ appear in a design, respectively. Under the BP model associated with $\mathcal{C}_2$, we can take $f_j = 2$ for all j to obtain a design given by Theorem 2.3, which we denote by $\mathcal{D}_{\text{cap}}$. An design given by Proposition 2.4 is $(f_0, \ldots, f_7) = (4,3,3,3,1,1,1,0)$, denoted by $\mathcal{D}_{\text{Rech}}$. By a complete computer search, the design $(f_0, \ldots, f_7) = (3,2,2,2,2,2,2,1)$ is $A_{\mathcal{C}_2}$-optimal among all competing designs, denoted by $\mathcal{D}_A$. And if we want the best efficiency only for the estimation of main effects, then $(f_0, \ldots, f_7) = (4,2,3,3,1,1,1,1)$ can minimize $\text{Var}(\hat{\theta}_1) + \text{Var}(\hat{\theta}_2) + \text{Var}(\hat{\theta}_3)$ among all competing designs, denote by $\mathcal{D}_{\text{main}}$.

Table 2.1 lists down all the variances of the least square estimators of the parameters in the model under these designs with $\sigma^2 = 1$. The $A_{\mathcal{C}_2}$-efficiency criterion and $\text{Var}(\hat{\theta}_1) + \text{Var}(\hat{\theta}_2) + \text{Var}(\hat{\theta}_3)$ are also computed. We can see that in terms of $A_{\mathcal{C}}$-optimality, $\mathcal{D}_{\text{Rech}}$ is worse than the other three, with relative efficiency being around 70% of the other three, although under $\mathcal{D}_{\text{Rech}}$ the main effects can be estimated almost as efficient as $\mathcal{D}_{\text{main}}$. It can be also noticed that $\mathcal{D}_{\text{cap}}$ is similar to $\mathcal{D}_A$ in terms $A_{\mathcal{C}}$-optimality, with relative efficiency being $5.50/5.69 \approx 97\%$. Even in terms of $\text{Var}(\hat{\theta}_i)$ $(i = 1,2,3)$, $\mathcal{D}_{\text{cap}}$ is still not far from $\mathcal{D}_A$, as we can find the relative efficiency is $0.673/0.750 \approx 90\%$. If we compare $\mathcal{D}_{\text{cap}}$ with $\mathcal{D}_{\text{main}}$, the relative efficiency is $1.69/2.25 \approx 75\%$. More or less, this numerical study illustrates the general performance of the baseline designs given by Theorem 2.3 and Proposition 2.4.

Table 2.1: The variances of the least square estimators under four designs.

	$\mathcal{D}_{\text{cap}}$	$\mathcal{D}_{\text{Rech}}$	$\mathcal{D}_A$	$\mathcal{D}_{\text{main}}$
$\text{Var}(\hat{\theta}_\phi)$	0.438	0.250	0.308	0.238
$\text{Var}(\hat{\theta}_1)$	0.750	0.583	0.673	0.521
$\text{Var}(\hat{\theta}_2)$	0.750	0.583	0.673	0.521
$\text{Var}(\hat{\theta}_3)$	0.750	0.583	0.673	0.646
$\text{Var}(\hat{\theta}_{12})$	1.000	1.917	1.058	1.238
$\text{Var}(\hat{\theta}_{13})$	1.000	1.917	1.058	1.282
$\text{Var}(\hat{\theta}_{23})$	1.000	1.917	1.058	1.282
$A_{\mathcal{C}_2}$-efficiency criterion	5.69	7.75	5.50	5.73
$\text{Var}(\hat{\theta}_1) + \text{Var}(\hat{\theta}_2) + \text{Var}(\hat{\theta}_3)$	2.25	1.75	2.02	1.69

In fact, $\mathcal{D}_{\text{cap}}$, $\mathcal{D}_{\text{Rech}}$, and $\mathcal{D}_A$ are all *balanced arrays*, the class of designs in which the number of appearance of a level combination g only depends on the number of ones g has. Balanced arrays were firstly introduced by Chakravarti [27], who used the name "partially balanced arrays". The simpler term "balanced array", which is more often used today, was later coined by Srivastava and Chopra [28]. As discussed in Stallings and Morgan [23], though out of the scope of this chapter, there is some evidence that balanced arrays will be highly efficient for estimating baseline effects. We refer readers to Morgan, Ghosh, and Dean [29] for an excellent review of balanced arrays.

2.5 Conclusion

We have derived a linear relationship between the OP and the BP. From its special pattern, we conclude that an echelon model has the same form under the two types of parameterization. We further discuss its implications for the estimability, optimality, and robustness of baseline designs. In particular, we show that certain orthogonal arrays continue to be optimal under the BP. We introduce general Rechtschaffner designs, showing they enjoy a robust property that is only available under the BP.

There are two possible future research directions. The first is illustrated by Example 2.5, in which we find the level permutations that minimize the bias caused by non-negligible effects. Under the main-effect-only model, this has been investigated by Mukerjee and Tang [4] and Li, Miller, and Tang [19]. However, it would be useful to obtain results for more general echelon models. The second is to consider a compromise between robust and optimal designs, which can be done by adding runs to a Rechtschaffner design. The compromise designs are expected to enjoy in-between performance in terms of both efficiency and robustness, as demonstrated for the main-effect model of Karunanayaka and Tang [30].

2.6 Proofs

Proof of Theorem 2.1. Let τ be a column vector with componenets $\tau_{\phi}, \tau_1, \tau_2, \tau_{12}, \ldots, \tau_{12\cdots m}$ in Yates order. Vectors θ and β are similarly defined. Let H_m be the m-fold Kronecker product of H and L_m the m-fold Kronecker product of L, where

$$H = \begin{bmatrix} 1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix} \text{ and } L = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

We then have $\beta = H_m\tau$ and $\tau = L_m\theta$. Therefore $\beta = H_mL_m\theta$ and $\theta = (H_mL_m)^{-1}\beta$. Theorem 1 follows by noting that H_mL_m is the m-fold Kronecker product of HL and $(H_mL_m)^{-1}$ is the m-fold Kronecker product of $(HL)^{-1}$ and the special forms of HL and $(HL)^{-1}$ as given by

$$HL = \begin{bmatrix} 1 & 1/2 \\ 0 & -1/2 \end{bmatrix} \text{ and } (HL)^{-1} = \begin{bmatrix} 1 & 1 \\ 0 & -2 \end{bmatrix}.$$

□

Proof of Theorem 2.2. This result follows immediately from Proposition 2.2. □

Proof of Corollary 2.2. For a design D, let D_π be the design obtained from D by level switching one or more factors. We use W and W_π to denote the model matrices under D and D_π for the BP, respectively. Matrices X and X_π are defined similarly for the OP. By Proposition 2.2, the ratio $(\det(X'X)/\det(W'W)) = (\det(X_\pi'X_\pi)/\det(W_\pi'W_\pi))$ is a constant which only depends on the model. Since $\det(X'X) = \det(X_\pi'X_\pi)$, we conclude that $\det(W'W) = \det(W_\pi'W_\pi)$. □

Proof of Theorem 2.3. Due to a result by Moriguti [5], $\mathrm{Var}(\hat{\beta}_v)$ attains its minimal value for each $v \in \mathcal{C}$ if $X_{\mathcal{C}}$ is orthogonal. If $\mathcal{C}$ is echelon, by Theorem 2.1 and Corollary 2.1, we have that $\theta_w = \sum_{v \supseteq w, v \in \mathcal{C}} c_v \beta_v$. If w is not contained by another u in $\mathcal{C}$, then $\theta_w = c_w \beta_w$. Thus, $\mathrm{Var}(\hat{\theta}_w) = c_w^2 \mathrm{Var}(\hat{\beta}_w)$ is minimized. □

Proof of Theorem 2.4. Consider the matrix $W_m = L_m$ in the proof of Theorem 2.1, which is the model matrix of the full model. Let W_m^* be the $N \times N$ submatrix of W_m, obtained by deleting all rows and columns except for the j_1-, j_2-,..., j_N-th rows and columns. It is sufficient to show that W_m^* is non-singular. Note that $j_1 = 1$ since a Rechtschaffner design always contains $x = (0, \ldots, 0)$ and the model always contains the intercept. The non-singularity of W_m^* is easily seen since W_m is a lower triangular matrix with all diagonals being one, which the case is because $W_m = W_{m-1} \otimes W_1$ and W_1 has the same pattern. □

Proof of Theorem 2.5. Let $\mathcal{C} = \{\zeta_0 = \phi, \zeta_1, \zeta_2, \ldots, \zeta_p\}$. Without loss of generality, let the i-th run $x_i = (x_{i1}, \ldots, x_{im})$ correspond to ζ_i, $i = 0, 1, \ldots, p$. The fitted model can be written as $E(Y) = W_{\mathcal{C}} \theta_{\mathcal{C}}$, where $E(Y) = (\tau_{\zeta_0}, \tau_{\zeta_1}, \ldots, \tau_{\zeta_p})'$ and $\theta_{\mathcal{C}} = (\theta_\phi, \theta_{\zeta_1}, \ldots, \theta_{\zeta_p})'$. Since there may exist some non-negligible effects θ_w with $w \notin \mathcal{C}$, we let the true model be $E(Y) = W_{\mathcal{C}} \theta_{\mathcal{C}} + \sum_{w \notin \mathcal{C}} W_w \theta_w$, where W_w is a $(p+1) \times 1$ column vector with the i-th entry equal to $\prod_{j \in w} g_{ij}$.

Let $\hat{\theta}_{\mathcal{C}}$ be the least square estimator from the fitted model. Then, $E(\hat{\theta}_{\mathcal{C}}) = (W_{\mathcal{C}}' W_{\mathcal{C}})^{-1} W_{\mathcal{C}}' E(Y) = \theta_{\mathcal{C}} + \sum_{w \notin \mathcal{C}} (W_{\mathcal{C}}' W_{\mathcal{C}})^{-1} W_{\mathcal{C}}' W_w \theta_w$. Thus, if we can show that for each $w \notin \mathcal{C}$, W_w is an all-zeros column vector, then the proof is completed. This is evident because $\prod_{j \in w} x_{ij}$ is one if s_i contains w as a subset, and zero otherwise. However, due to the fact that $\mathcal{C}$ is echelon, no s_i can contain w as a subset. □

Proof of Proposition 2.4. Let model (2.6) under the Rechtschaffner design $D_{\mathcal{C}}$ (i.e., $f_j = 1$ for $j = 0, 1, \ldots, p$.) be $E(Y) = W_{\mathcal{C}} \theta_{\mathcal{C}}$, where $E(Y) = (\tau_{\zeta_0}, \tau_{\zeta_1}, \ldots, \tau_{\zeta_p})'$ and $\theta_{\mathcal{C}} = (\theta_\phi, \theta_{\zeta_1}, \ldots, \theta_{\zeta_p})'$. Consider an N-run Rechtschaffner design and let E be the $(p+1) \times (p+1)$ identity matrix. The model matrix can be written as $A W_{\mathcal{C}}$, where A is an $N \times (p+1)$ matrix. The first f_0 rows of A are the first row of E, the following f_1 rows are the second row of E, and so on. The $A_{\mathcal{C}}$-efficiency is

$$\operatorname{tr}\left((AW_{\mathcal{C}})'(AW_{\mathcal{C}})\right)^{-1} = \operatorname{tr}\left(W_{\mathcal{C}}^{-1}(A'A)^{-1}(W_{\mathcal{C}}')^{-1}\right) = \operatorname{tr}\left((A'A)^{-1}(W_{\mathcal{C}}')^{-1}(W_{\mathcal{C}})^{-1}\right)$$

It is evident that $(A'A)^{-1} = diag(f_0^{-1}, f_1^{-1}, \ldots, f_p^{-1})$, so the $A_{\mathcal{C}}$-efficiency is $\sum_{j=0}^{p} q_j f_j^{-1}$, where q_j is the (j,j)-th element of $(W_{\mathcal{C}}')^{-1}(W_{\mathcal{C}})^{-1}$, for $j = 0, 1, \ldots, p$. By Cauchy-Schwarz inequality, subject to $\sum_{j=0}^{p} f_j = N$, $\sum_{j=0}^{p} q_j f_j^{-1}$ is minimized if $f_j = N\left({q_j}^{0.5}/\sum_{j=0}^{p} {q_j}^{0.5}\right)$, so the proof can be completed by showing q_j is the number of subsets in $\mathcal{C}$ that contain ζ_j.

By definition (2.2), for any $w \in \mathcal{C}$, $\theta_w = \sum_{u \subseteq w} \tau_u (-1)^{|w|-|u|}$, which is equal to $\sum_{u \in \mathcal{C}, u \subseteq w} \tau_u (-1)^{|w|-|u|}$ since $\mathcal{C}$ is echelon. It is then implied that $\theta_{\mathcal{C}} = W_{\mathcal{C}}^{-1} E(Y)$ gives the definition back, and thus the j-th column of $W_{\mathcal{C}}^{-1}$ is

$$\Big(\ (-1)^{|\zeta_0|-|\zeta_j|} I(\zeta_0 \supseteq \zeta_j), (-1)^{|\zeta_1|-|\zeta_j|} I(\zeta_1 \supseteq \zeta_j), \ldots, (-1)^{|\zeta_p|-|\zeta_j|} I(\zeta_p \supseteq \zeta_j)\ \Big)',$$

where $I(\zeta_i \supseteq \zeta_j) = 1$ if ζ_i contains ζ_j as a subsets, and 0 otherwise. Now we can find that the (j,j)th element of $(W_{\mathcal{C}}')^{-1}(W_{\mathcal{C}})^{-1}$, which is the squared length of the jth column vector of $W_{\mathcal{C}}^{-1}$, is $\sum_{i=0}^{p} \{(-1)^{|\zeta_i|-|\zeta_j|} I(\zeta_i \supseteq \zeta_j)\}^2 = \sum_{i=0}^{p} I(\zeta_i \supseteq \zeta_j)$ $(j = 0, \ldots, p)$. □

Chapter 3

Uniform projection designs and strong orthogonal arrays

3.1 Introduction

Computer experiments are routinely used in scientific and engineering research nowadays. In a computer experiment, as the true model is usually characterized by highly complex differential equations, the experimenter needs to build a surrogate model that is easy to compute, based on the experimental data from running the computer code that represents the true model. To capture the overall pattern of the true model, the design points should be spread out as uniformly as possible throughout the experimental region. This space-filling idea has appeared early in Box and Draper [31] and later in Sacks and Ylvisaker [32], who indicated that uniformly spreading design points can minimize the bias of the fitted model.

In the context of computer experiments, the rationale for using space-filling designs is intuitive as one would like to have information in every portion of the design region when the true model as represented by a computer code is very complex. As shown by Vazquez and Bect [33], space-filling designs can also be theoretically justified in terms of their performances in the mean squared prediction error. We refer to Chapter 5 of Santner, Williams, Notz, and Williams [34] for a full discussion on the need for space-filling designs. When a computer model involves a large number of factors, it is sensible to first conduct a sensitivity analysis to identify the important factors and then build a surrogate statistical model based on these important factors. In such circumstances, more relevant are the designs that are space-filling in low dimensional projections. Broadly speaking, finding space-filling designs can be done either via computational algorithms or by theoretical construction. The focus of this chapter is on theoretical construction.

One way to measure the uniformity of the design points is to use a distance or a discrepancy criterion, which can be then optimized, sometimes analytically but most often via computational algorithms. For example, by maximizing the minimum distance between design points, maximin distance designs are obtained, and such designs enjoy an asymptotic optimality property as shown in Johnson, Moore, and Ylvisaker [8]. As for the discrepancy criteria, the centered L_2-discrepancy proposed by Hickernell [9] seems to be the most popular one. Recently, Sun, Wang, and Xu [13] proposed and investigated a new criterion, called the uniform projection criterion, which computes the centered L_2-discrepancy for every two-dimensional projection of a design and then takes the average. Their empirical study indicates that by minimizing the uniform projection criterion, the resulting designs are robust under different criteria and different types of model, making this new criterion attractive to experimenters.

An alternative approach does not compute a numerical measure, and is related to orthogonal arrays. Work along this line goes back to the well-known Latin hypercube designs proposed by McKay, Beckman, and Conover [12], which are orthogonal arrays of strength one. Latin hypercube designs promise stratification in all univariate projections, and their usefulness for computer experiments has been widely recognized. Later, using orthogonal arrays of strength $t \geq 2$, Owen [35] and Tang [36] independently gave constructions of space-filling designs that achieve stratification in all t and lower dimensional projections. He and Tang [37] proposed strong orthogonal arrays, a class of designs that are more space-filling than comparable ordinary orthogonal arrays. Focusing on two dimensional projections, He, Cheng, and Tang [14] introduced strong orthogonal arrays of strength 2+ and presented construction results for such designs.

In this chapter, we explore the connections between uniform projection designs as proposed in Sun, Wang, and Xu [13] and strong orthogonal arrays of strength 2+ as introduced in He, Cheng, and Tang [14]. Both of these classes of designs are space-filling in two dimensional projections but are motivated from very different angles. A key finding in our studies is a decomposition of the centered L_2-discrepancy for a two-factor design into a sum of squares where each square measures one aspect of uniformity of design points. This leads to a decomposition of the uniform projection criterion into two terms, with one term measuring the discrepancy between a given design and a strong orthogonal array of 2+ and the other term representing the residual discrepancy. We then show through a number of results that strong orthogonal arrays of strength 2+ are optimal or nearly optimal under the uniform projection criterion.

The chapter is organized as follows. Section 3.2 introduces the notation and background. Section 3.3 gives a new expression of the centered L_2-discrepancy for two-factor designs, which leads to a useful decomposition of the uniform projection criterion. In Section 3.4, we utilize the decomposition to study the performance of strong orthogonal arrays of 2+ under the uniform projection criterion. The chapter is summarized by Section 3.5, and all proofs are given in Section 3.6.

3.2 Notation and background

An orthogonal array of strength t, denoted by $\mathrm{OA}(N, m, s_1 \times \cdots \times s_m, t)$, is an $N \times m$ matrix in which the jth column has s_j levels from $\{0, 1, \ldots, s_j - 1\}$ such that, for any t columns, all possible level-combinations appear equally often. If $s_1 = \cdots = s_m = s$, the array becomes symmetric and is denoted by $\mathrm{OA}(N, m, s, t)$. A general s-level factorial design of N runs and m factors is represented by an $N \times m$ matrix $D = [x_{ij}]$, where $x_{ij} \in \mathbb{Z}_s = \{0, 1, \ldots, s-1\}$, and each row corresponds to an experimental run and each column to a factor. Design D is called balanced if it is an $\mathrm{OA}(N, m, s, 1)$. Note that a balanced orthogonal array is not a "balanced array" we introduced earlier in the previous chapter. We use D_i to denote the ith column of D and let $D_{ij} = (D_i, D_j)$. Therefore, D_i and D_{ij} are one- and two-dimensional projection designs of D, respectively. For a non-trivial divisor α of s, let $\lfloor D_i/\alpha \rfloor$ be the column obtained by collapsing the s levels of D_i into s/α levels via the mapping $x \to \lfloor x/\alpha \rfloor$, where $\lfloor x \rfloor$ is the integer part of x.

Definition 3.1. *Let $s = \alpha s'$. An $N \times m$ matrix D with entries from $\mathbb{Z}_s$ is a strong orthogonal array of strength 2+, denoted by $\mathrm{SOA}_\alpha(N, m, s, 2+)$, if for any $i \neq j$, $(\lfloor D_i/\alpha \rfloor, D_j)$ is an $\mathrm{OA}(N, 2, s' \times s, 2)$.*

Clearly, an $\mathrm{SOA}_\alpha(N, m, s, 2+)$ is a design that stratifies on the $s \times s'$ and $s' \times s$ grids in every two-dimensional projection, and thus $N = \lambda s s'$ for some integer λ. We call λ the index of the array. An example of $\mathrm{SOA}_2(16, 10, 4, 2+)$ is given below, which is presented in the transposed form:

$$\begin{pmatrix}
2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 3 & 3 & 3 & 3 \\
2 & 2 & 0 & 0 & 2 & 2 & 0 & 0 & 1 & 1 & 3 & 3 & 1 & 1 & 3 & 3 \\
2 & 0 & 2 & 0 & 2 & 0 & 2 & 0 & 1 & 3 & 1 & 3 & 1 & 3 & 1 & 3 \\
2 & 2 & 0 & 0 & 1 & 1 & 3 & 3 & 2 & 2 & 0 & 0 & 1 & 1 & 3 & 3 \\
2 & 0 & 2 & 0 & 1 & 3 & 1 & 3 & 2 & 0 & 2 & 0 & 1 & 3 & 1 & 3 \\
2 & 0 & 1 & 3 & 2 & 0 & 1 & 3 & 2 & 0 & 1 & 3 & 2 & 0 & 1 & 3 \\
0 & 1 & 2 & 3 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 0 & 1 & 2 & 3 \\
0 & 2 & 1 & 3 & 2 & 0 & 3 & 1 & 2 & 0 & 3 & 1 & 0 & 2 & 1 & 3 \\
0 & 2 & 2 & 0 & 1 & 3 & 3 & 1 & 2 & 0 & 0 & 2 & 3 & 1 & 1 & 3 \\
0 & 2 & 2 & 0 & 2 & 0 & 0 & 2 & 1 & 3 & 3 & 1 & 3 & 1 & 1 & 3
\end{pmatrix}. \tag{3.1}$$

It can be verified that, for example, $(\lfloor D_1/2 \rfloor, D_2)$ is an $\mathrm{OA}(16, 2, 2 \times 4, 2)$, meaning that D_{12} stratifies on the 2×4 grid, where each cell of the grid contains $\lambda = 2$ points. One advantage of strong orthogonal arrays over ordinary orthogonal arrays with the same number of levels is the larger number of factors the former can accommodate. Suppose sixteen experimental runs are allowed. An $\mathrm{OA}(16, m, 4, 2)$ achieves stratification on the 4×4 grid in every two dimensions, but it only allows us to study at most five factors as an $\mathrm{OA}(16, 5, 4, 2)$ exists but an $\mathrm{OA}(16, 6, 4, 2)$ does not. In contrast, an $\mathrm{SOA}_2(16, m, 4, 2+)$ promises the stratification on the 4×2 and 2×4 grids in every two dimensions, and allows us to study up to ten factors, which is twice more. Another advantage is the run size flexibility. To have an $\mathrm{OA}(N, m, 4, 2)$, N needs to be a multiple of 16, but there exists $\mathrm{SOA}_2(N, m, 4, 2+)$ with $N = 8$ or 24. For more details on strong orthogonal arrays of strength 2+, we refer to He, Cheng, and Tang [14].

The centered L_2-discrepancy proposed by Hickernell [9] is a popular criterion to measure the uniformity of a design. For an s-level $N \times m$ design $D = [x_{ij}]$, its (squared) centered

L_2-discrepancy can be evaluated as below:

$$
\begin{aligned}
\mathrm{CD}(D) = &-\frac{2}{N}\sum_{i=1}^{N}\prod_{k=1}^{m}\left(1+\frac{1}{2}|z_{ik}|-\frac{1}{2}|z_{ik}|^2\right)\\
&+\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\prod_{k=1}^{m}\left(1+\frac{1}{2}|z_{ik}|+\frac{1}{2}|z_{jk}|-\frac{1}{2}|z_{ik}-z_{jk}|\right)+\left(\frac{13}{12}\right)^m,
\end{aligned}
\tag{3.2}
$$

where $z_{ik} = (2x_{ik} + 1 - s)/(2s)$. Designs having lower $\mathrm{CD}(D)$ values are more space-filling, since $\mathrm{CD}(D)$ measures the discrepancy between the empirical distribution of the design points and the uniform distribution. If we apply $\mathrm{CD}(\cdot)$ to every two-dimensional projection of D and take the average, the uniform projection criterion [see 13] is obtained. That is,

$$
\phi(D) = \frac{2}{m(m-1)}\sum_{i<j}\mathrm{CD}(D_{ij}). \tag{3.3}
$$

A design minimizes $\phi(D)$ among all balanced competing designs is called a uniform projection design. The uniform projection criterion only considers the two-dimensional projections, since the uniformity in low dimensions is usually considered more important than that in high dimensions. Interestingly, an empirical study in Sun, Wang, and Xu [13] indicates that even when projected onto a t-dimensional space for any $t > 2$, a uniform projection design performs well under several different criteria, including the maximin distance, orthogonality, and the maximum projection criterion [see 11].

We conclude this section with Proposition 3.1, which states that an orthogonal arrays of strength two is a uniform projection design, and gives a lower bound on the uniform projection criterion. We point out here that Proposition 3.1 can be easily obtained by combining Theorems 1 and 2 in Ma, Fang, and Lin [38].

Proposition 3.1. *For any balanced s-level $N \times m$ design D, we have that $\phi(D) \geq O_s$ and that the equality holds if and only if D is an $\mathrm{OA}(N, m, s, 2)$, where*

$$
O_s = \begin{cases} 13/(72s^2) - 1/(144s^4) & \text{when } s \text{ is odd,}\\ 13/(72s^2) + 7/(288s^4) & \text{when } s \text{ is even.}\end{cases}
$$

Finally, we point out that, in the context of computer experiments, the number s of levels needs to be of at least moderate size in order to fit response surfaces that have multiple extrema.

3.3 A decomposition of uniform projection criterion

In this section, we derive a new expression of the centered L_2-discrepancy for two-factor designs. This expression not only offers an insight into the centered L_2-discrepancy in (3.2) but also leads to a decomposition of the uniform projection criterion $\phi(D)$ in (3.3). The decomposition provides an important link of the uniform projection criterion with strong orthogonal arrays of strength 2+, and the usefulness of which will be further explored in the next section. For $i, j \in \mathbb{Z}_s$, we define

$$\begin{aligned} b_i &= \frac{1}{2}\left|\frac{2i+1-s}{2s}\right| - \frac{1}{2}\left|\frac{2i+1-s}{2s}\right|^2, \\ a_{ij} &= \frac{1}{2}\left|\frac{2i+1-s}{2s}\right| + \frac{1}{2}\left|\frac{2j+1-s}{2s}\right| - \frac{1}{2}\left|\frac{i-j}{s}\right|. \end{aligned} \tag{3.4}$$

Let $\Omega_s = \mathbb{Z}_s \times \mathbb{Z}_s$. For a level combination $x = (x_1, x_2) \in \Omega_s$, define $B_x = b_{x_1}b_{x_2}$. For a pair of level combinations $x = (x_1, x_2)$ and $y = (y_1, y_2)$ both in Ω_s, not necessarily distinct, define $A_{xy} = a_{x_1y_1}a_{x_2y_2}$. The following proposition is implied by the results of Ma, Fang, and Lin [38], and can also be easily verified directly using the definition of $\mathrm{CD}(D)$ in (3.2) for $m = 2$ in conjunction with the definitions of B_x and A_{xy} through the b_is and a_{ij}s in (3.4).

Proposition 3.2. *Let D be a balanced s-level $N \times 2$ design. For a level combination $x = (x_1, x_2) \in \Omega_s$, let f_x be the number of times x appears in D. We have that*

$$\mathrm{CD}(D) = -\frac{2}{N}\sum_{x\in\Omega_s} f_x B_x + \frac{1}{N^2}\sum_{x\in\Omega_s}\sum_{y\in\Omega_s} f_x f_y A_{xy} + C_s, \tag{3.5}$$

where C_s is a constant determined by s.

Based on Proposition 3.2, we prove Theorem 3.1, which plays a key role in most of our theoretical results.

Theorem 3.1. *Let D be a balanced s-level $N \times 2$ design and define $v_x = f_x - N/s^2$. For a positive integer $s > 2$, let $s_1 = \lfloor s/2 \rfloor - 1$ and $s_2 = \lceil s/2 \rceil$, where $\lceil x \rceil$ stands for the smallest integer not less than x. We have that*

$$\begin{aligned} \mathrm{CD}(D) &= O_s + \frac{1}{N^2} \sum_{x \in \Omega_s} \sum_{y \in \Omega_s} v_x v_y A_{xy} \\ &= O_s + \frac{1}{(Ns)^2} \left\{ \sum_{i=0}^{s_1} \sum_{j=0}^{s_1} \delta_{ij} \left(\sum_{x_1 \le i, x_2 \le j} v_x \right)^2 + \sum_{i=s_2}^{s-1} \sum_{j=0}^{s_1} \delta_{ij} \left(\sum_{x_1 \ge i, x_2 \le j} v_x \right)^2 \right. \\ &\quad \left. + \sum_{i=0}^{s_1} \sum_{j=s_2}^{s-1} \delta_{ij} \left(\sum_{x_1 \le i, x_2 \ge j} v_x \right)^2 + \sum_{i=s_2}^{s-1} \sum_{j=s_2}^{s-1} \delta_{ij} \left(\sum_{x_1 \ge i, x_2 \ge j} v_x \right)^2 \right\}, \end{aligned} \tag{3.6}$$

where O_s is given in Proposition 3.1 and δ_{ij} is defined as follows. If s is odd, then $\delta_{ij} \equiv 1$. If s is even, then

$$\delta_{ij} = \begin{cases} 1/4 & \text{if } i, j \in \{s_1, s_2\}; \\ 1/2 & \text{if only one of } i \text{ and } j \in \{s_1, s_2\}; \\ 1 & \text{if } i, j \notin \{s_1, s_2\}. \end{cases}$$

Theorem 3.1 may look a bit complicated but actually has a very intuitive interpretation. It shows that the centered L_2-discrepancy for a two-factor s-level balanced design can be written as a constant O_s plus a weighted sum of squares, where each square of form $(\sum_x v_x)^2$ corresponds to a rectangle area with one vertex being one of the four corners in the $s \times s$ grid. These square terms are all equal to 0 if $v_x = 0$ for all x which happens if and only f_x is a constant, which amounts to that the design points are uniformly distributed on the $s \times s$ grid and the design is an OA$(N, 2, s, 2)$. Each term of form $(\sum_x v_x)^2$ in Theorem 3.1, therefore, provides a measure of uniformity of the design within a rectangle region formed using one of four corners in the $s \times s$ grid as a vertex. Two illustrative examples are given below.

Example 3.1. *Take $s = 4$ in Theorem 3.1. Then $s_1 = 1$ and $s_2 = 2$. If we define*

$$\begin{aligned}
V_{2\times2} =& \left(v_{(0,0)} + v_{(0,1)} + v_{(1,0)} + v_{(1,1)}\right)^2 + \left(v_{(0,2)} + v_{(0,3)} + v_{(1,2)} + v_{(1,3)}\right)^2 \\
&+ \left(v_{(2,0)} + v_{(2,1)} + v_{(3,0)} + v_{(3,1)}\right)^2 + \left(v_{(2,2)} + v_{(2,3)} + v_{(3,2)} + v_{(3,3)}\right)^2, \\
V_{1\times2} =& \left(v_{(0,0)} + v_{(0,1)}\right)^2 + \left(v_{(0,2)} + v_{(0,3)}\right)^2 + \left(v_{(3,0)} + v_{(3,1)}\right)^2 + \left(v_{(3,2)} + v_{(3,3)}\right)^2, \\
V_{2\times1} =& \left(v_{(0,0)} + v_{(1,0)}\right)^2 + \left(v_{(0,3)} + v_{(1,3)}\right)^2 + \left(v_{(2,0)} + v_{(3,0)}\right)^2 + \left(v_{(2,3)} + v_{(3,3)}\right)^2, \\
V_{1\times1} =& v_{(0,0)}^2 + v_{(0,3)}^2 + v_{(3,0)}^2 + v_{(3,3)}^2,
\end{aligned}$$

then we have that $\mathrm{CD}(D) = O_4 + (4N)^{-2}\left(4^{-1}V_{2\times2} + 2^{-1}V_{2\times1} + 2^{-1}V_{1\times2} + V_{1\times1}\right)$. Note that each $V_{k\times l}$ collects four square terms, corresponding to four $k \times l$ sub-grids of the 4×4 grid with each sub-grid starting from one of the four corners of the 4×4 grid. If D is an $\mathrm{OA}(N, 2, 4, 2)$, then $V_{2\times2} = V_{2\times1} = V_{1\times2} = V_{1\times1} = 0$ as $v_x = 0$ for all x. Now suppose that D is an $\mathrm{SOA}_2(N, 2, 4, 2+)$, then we can easily see that $V_{2\times2} = V_{2\times1} = V_{1\times2} = 0$ as an $\mathrm{SOA}_2(N, 2, 4, 2+)$ achieve stratifications on 2×2, 4×2 and 2×4 grids. We therefore have that $\mathrm{CD}(D) = O_4 + \psi(D) + \epsilon(D)$, where $\psi(D) + \epsilon(D)$ represents a total discrepancy of the design from an $\mathrm{OA}(N, 2, 4, 2)$, $\psi(D) = (4N)^{-2}\left(4^{-1}V_{2\times2} + 2^{-1}V_{2\times1} + 2^{-1}V_{1\times2}\right)$ measures the discrepancy of the design from an $\mathrm{SOA}_2(N, 2, 4, 2+)$ and $\epsilon(D) = (4N)^{-2}V_{1\times1}$ is the residual discrepancy of the design from an $\mathrm{OA}(N, 2, 4, 2)$. This decomposition is analogous to the decomposition of the total sum of squares into a regression sum of square plus a residual sum of squares in regression analysis.

Example 3.2. *Let $s = 9$ in Theorem 3.1. Then we have $s_1 = 3$ and $s_2 = 5$. Similar to Example 3.1, we can define $V_{k\times l}$ for $k, l = 1, 2, 3, 4$, where each $V_{k\times l}$ collects four square terms, corresponding to four $k \times l$ sub-grids starting from the four corners. For example,*

$$V_{4\times2} = \left(\sum_{x_1\le3, x_2\le1} v_x\right)^2 + \left(\sum_{x_1\ge5, x_2\le1} v_x\right)^2 + \left(\sum_{x_1\le3, x_2\ge7} v_x\right)^2 + \left(\sum_{x_1\ge5, x_2\ge7} v_x\right)^2.$$

In this case, (3.6) can be written as

$$\mathrm{CD}(D) = O_9 + (9N)^{-2}\sum_{k=1}^{4}\sum_{l=1}^{4} V_{k\times l}.$$

Let $\psi(D) = (9N)^{-2}\sum_{(k,l)\in S_1} V_{k\times l}$ and $\epsilon(D) = (9N)^{-2}\sum_{(k,l)\in S_2} V_{k\times l}$, where $S_1 = \{(k,l) :$ at least one k or l is a multiple of $3\}$ and $S_2 = \{(k,l) :$ neither k nor l is a multiple of $3\}$. Then, $\mathrm{CD}(D) = O_9 + \psi(D) + \epsilon(D)$, where $\psi(D) \geq 0$ represents the discrepancy of the design from an $\mathrm{SOA}_3(N, 2, 9, 2+)$ and is equal to 0 if D is an $\mathrm{SOA}_3(N, 2, 9, 2+)$, and $\epsilon(D) \geq 0$ is the residual discrepancy of design D from an $\mathrm{OA}(N, 2, 9, 2)$.

In general, let $s = \alpha s'$ and D be a balanced s-level $N \times m$ design. Consider a projection design D_{ij} given by the ith and jth columns of D. Following the same ideas in Examples 3.1 and 3.2, we define $V_{k\times l}$ for $k, l = 1, \ldots, s_1 + 1$, which is a sum of four squares in (3.6), each corresponding to a $k \times l$ sub-grid from a corner. More precisely, for $k, l = 1, \ldots, s_1 + 1$, define

$$V_{k\times l} = \left(\sum_{x_1=0}^{k-1}\sum_{x_2=0}^{l-1} v_x\right)^2 + \left(\sum_{x_1=0}^{k-1}\sum_{x_2=s-l}^{s-1} v_x\right)^2 + \left(\sum_{x_1=s-k}^{s-1}\sum_{x_2=0}^{l-1} v_x\right)^2 + \left(\sum_{x_1=s-k}^{s-1}\sum_{x_2=s-l}^{s-1} v_x\right)^2.$$

We then obtain

$$\mathrm{CD}(D_{ij}) = O_s + (sN)^{-2}\sum_{k=1}^{s_1+1}\sum_{l=1}^{s_1+1} q_{kl}V_{k\times l}, \tag{3.7}$$

where q_{kl} is a δ_{ij} in Theorem 3.1, taking on one of the three values: 1/4, 1/2, or 1. Decompose the double summation above into two parts, denoted by $\psi(D_{ij})$ and $\epsilon(D_{ij})$, where $(sN)^{-2}q_{kl}V_{k\times l}$ belongs to $\psi(D_{ij})$ if at least one of k and l is a multiple of α, and belongs to $\epsilon(D_{ij})$ otherwise. Then we have that $\mathrm{CD}(D_{ij}) = O_s + \psi(D_{ij}) + \epsilon(D_{ij})$. Define $\Psi(D) = 2/\{m(m-1)\}\sum_{i<j}\psi(D_{ij})$ and $E(D) = 2/\{m(m-1)\}\sum_{i<j}\epsilon(D_{ij})$. Theorem 3.2 below is obtained.

Theorem 3.2. *Let $s = \alpha s'$. For a balanced s-level $N \times m$ design D, we have*

$$\phi(D) = O_s + \Psi(D) + E(D),$$

where $\phi(D) - O_s = \Psi(D) + E(D)$ is a total discrepancy of design D from an $\mathrm{OA}(N, m, s, 2)$, $\Psi(D) \geq 0$ is a discrepancy of D from an $\mathrm{SOA}_\alpha(N, m, s, 2+)$, and $E(D) \geq 0$ is the residual discrepancy of D from an $\mathrm{OA}(N, m, s, 2)$. Moreover, $\Psi(D) = 0$ if and only if D is an $\mathrm{SOA}_\alpha(N, m, s, 2+)$ and $\Psi(D) = E(D) = 0$ if and only if D is an $\mathrm{OA}(N, m, s, 2)$.

Theorems 3.1 and 3.2 provide insights into the uniform projection criterion. Theorem 3.2 also provides a big-picture justification for strong orthogonal arrays of strength 2+. More specific results on the performance of strong orthogonal arrays of strength 2+ under the uniform projection criterion are presented in the next section.

3.4 Optimality or near optimality of strong orthogonal arrays of strength 2+

Proposition 3.1 says that an $\mathrm{OA}(N, m, s, 2)$ is a uniform projection design. However, an $\mathrm{OA}(N, m, s, 2)$ may not exist; and even if it exists, the run size N may be too large for the experimenter to afford. In such situations, $\mathrm{SOA}_\alpha(N, m, s, 2+)$s provide an attractive class of alternatives. Theorem 3.2 shows that an $\mathrm{SOA}_\alpha(N, m, s, 2+)$ minimizes a component of the uniform projection criterion. In this section, we will show that, under the uniform projection criterion, an $\mathrm{SOA}_\alpha(N, m, s, 2+)$ is nearly optimal in general and is optimal for a special case.

Theorem 3.3. *Consider the decomposition in Theorem 3.2. If D is an $\mathrm{SOA}_\alpha(N, m, s, 2+)$, then $\Psi(D) = 0$ and $E(D) \leq k_\alpha/s^4$ for a constant k_α determined only by α. Therefore, $\phi(D) \leq O_s + k_\alpha/s^4$.*

Most useful are $\mathrm{SOA}_\alpha(N, m, s, 2+)$s with small α values due to their economical run sizes. For $\alpha = 2, 3$ and 4, we have found that k_α is equal to 1/4, 10/9, and 13/4, respectively. More discussion on the meaning and value of k_α will be provided in Section 3.5. For a balanced s-level $N \times m$ designs D, we define its ϕ-efficiency as $\phi_{\text{eff}}(D) = O_s/\phi(D)$, which is a simple measure of the goodness of D under $\phi(D)$. If $\phi_{\text{eff}}(D)$ is close to 1, then D performs well under $\phi(D)$. Corollary 3.1 immediately follows from Theorem 3.3.

Corollary 3.1. *If D is an $\mathrm{SOA}_\alpha(N, m, s, 2+)$, then $\phi_{\text{eff}}(D) \geq O_s/(O_s + k_\alpha/s^4)$.*

The expression of O_s in Proposition 3.1 shows that $O_s = O(1/s^2)$, which implies that the lower bound on the ϕ-efficiency $O_s/(O_s + k_\alpha/s^4)$ converges to 1 as $s \to \infty$. Corollary 3.1, therefore, establishes near optimality of $\mathrm{SOA}_\alpha(N, m, s, 2+)$s under the uniform projection criterion.

The lower bound in Corollary 3.1 is calculated and the results are given in Table 3.1 under the heading of 'any λ' for $\alpha = 2$ and 3 and $s \leq 18$. We see that all entries except one in Table 3.1 are greater than 90%. Those entries under '$\lambda = 1$' are obtained from improved lower bounds on $\phi_{\text{eff}}(D)$ to be presented in the next two results. The first of these establishes optimality of an $\text{SOA}_2(N, m, s, 2+)$ with index $\lambda = 1$ under $\phi(D)$.

Theorem 3.4. *An* $\text{SOA}_2(s^2/2, m, s, 2+)$ *minimizes* $\phi(D)$ *among all balanced competing designs of* s *levels,* $N = s^2/2$ *runs and* m *factors.*

It is worth investigating that whether $\phi(D)$ is minimized only by an $\text{SOA}_2(s^2/2, m, s, 2+)$, but we will leave this question to our research work in the future. The construction of $\text{SOA}_2(s^2/2, m, s, 2+)$s can be found in He, Cheng, and Tang [14]. Some examples are an $\text{SOA}_2(18, 4, 6, 2+)$, an $\text{SOA}_2(32, 5, 8, 2+)$, and an $\text{SOA}_2(50, 6, 10, 2+)$. The basic idea in the proof of Theorem 3.4, which is given in the Appendix, is to derive and use a better lower bound on $\phi(D)$ than O_s in combination with Theorem 3.3. A similar argument also leads to a better lower bound on $\phi(D)$ than O_s when $\alpha = 3$ and $\lambda = 1$, which results in an improved lower bound on the ϕ_{eff}. This is presented in the following result.

Theorem 3.5. *If* D *is an* $\text{SOA}_3(s^2/3, m, s, 2+)$, *then* $\phi_{\text{eff}}(D) \geq \{O_s + 4/(9s^4)\}/\{O_s + 10/(9s^4)\}$.

Though an optimality property like Theorem 3.4 cannot be obtained, it can be seen in Table 3.1 that $\text{SOA}_3(s^2/3, m, s, 2+)$s perform well under $\phi(D)$. He, Cheng, and Tang [14] constructed an $\text{SOA}_3(27, 6, 9, 2+)$, an $\text{SOA}_3(48, 5, 12, 2+)$, and an $\text{SOA}_3(75, 6, 15, 2+)$, and all these designs have ϕ_{eff} values larger than 95%.

When s is small and m/N is large, the ϕ-efficiency defined by $LB/\phi(D)$ in Sun, Wang, and Xu [13] may be preferred, where

$$LB = \frac{5m(4s^4 + 2(13N - 17)s^2 - N + 5) - (N - 1)(8s^4 + 150s^2 - 33)}{720(m - 1)(N - 1)s^4} + \frac{1 + (-1)^s}{64s^4}.$$

For example, by Theorem 3.3, we have that $\phi_{\text{eff}}(D) \geq LB/(O_4 + 1/1024) = 95.3\%$, for any D being an $\text{SOA}_2(16, 10, 4, 2+)$, which improves the result shown in Table 3.1. In fact, the $\text{SOA}_2(16, 10, 4, 2+)$ given in (3.1) actually has a ϕ-efficiency 97.9%.

Table 3.1: The lower bounds of the ϕ-efficiency for SOAs(2+) with $\alpha = 2$ and 3.

$\alpha = 2$			$\alpha = 3$		
s	any λ	$\lambda = 1$	s	any λ	$\lambda = 1$
4	92.1%	100%	6	85.4%	91.3%
6	96.3%	100%	9	92.9%	95.8%
8	97.9%	100%	12	95.9%	97.5%
10	98.6%	100%	15	97.3%	98.4%
12	99.0%	100%	18	98.1%	98.9%

3.5 Concluding remarks

A brief summary of this chapter is as follows. We first derive a new expression for the centered L_2-discrepancy for two-factor designs, which shows that the centered L_2-discrepancy can be written as a sum of squares with each square measuring how closely a design resembles an OA$(N, 2, s, 2)$ within a rectangle area from a corner in the $s \times s$ grid. Besides being insightful, this expression leads to a decomposition of the uniform projection criterion as given by $\phi(D) = O_s + \Psi(D) + E(D)$, where O_s can be interpreted as the base discrepancy, $\Psi(D)$ represents a discrepancy of design D from an SOA$_\alpha(N, m, s, 2+)$, and $E(D)$ is the residual discrepancy from an OA$(N, m, s, 2)$. This decomposition then allows us to establish that SOA$_\alpha(N, m, s, 2+)$s are nearly optimal in general and optimal in a special case, under the uniform projection criterion $\phi(D)$.

Let F_α be a design that maximizes the centered L_2-discrepancy among all balanced α-level $\alpha \times 2$ designs. As detailed in the proof of Theorem 3.3 in the Appendix, the k_α in Theorem 3.3 can be obtained by $k_\alpha = \alpha^4\left(\mathrm{CD}(F_\alpha) - O_\alpha\right)$. We conjecture that for any α, the design consisting of $\{(0,0), (1,1), \ldots, (\alpha-1, \alpha-1)\}$ is an F_α, which we have verified for $\alpha = 2, \ldots, 7$. Though not of much value to the theme of this chapter, settling the conjecture would be an interesting pursuit for an inquiring mind.

Sun, Wang, and Xu [13] showed that L_1-equidistant designs are uniform projection designs. As equidistant designs are maximin distance designs, given the connections between strong orthogonal arrays and uniform projection designs, one would expect that strong orthogonal arrays also perform well under some distance-based criterion. It would be interesting to conduct some investigation along this direction.

The results in this chapter are concerned with the space-filling properties in two dimensional projections. It would be of great interest to obtain some similar results for designs that are space-filling in $t \geq 3$ dimensional projections. A uniform projection criterion for t dimensions can be easily defined by taking average of the centered L_2-discrepancies of all t dimensional projection designs. Orthogonal arrays of strength t are space-filling in all t dimensions. Strong orthogonal arrays of strength t are as space-filling as comparable orthogonal arrays of strength t in all t dimensions but more space-filling than the latter in all g dimensions where $g < t$. For $t \geq 4$, orthogonal arrays of strength t require large run sizes and strong orthogonal arrays of strength t are even more expensive. But for $t = 3$, the run sizes of both types of arrays are reasonable. This is the case we will focus on in our future studies.

3.6 Proofs

Proof of Theorem 3.1. An unlabeled result in Ma, Fang, and Lin [38] which is given right before their equation (3.7) implies the following identities:

$$\begin{aligned}
&\sum_{y\in\Omega_s} A_{xy} = s^2 B_x && \text{if } s \text{ is odd;} \\
&\sum_{y\in\Omega_s} A_{xy} = s^2 B_x + \frac{1}{8}(b_{x_1} + b_{x_2}) + \frac{1}{64}s^2 && \text{if } s \text{ is even.}
\end{aligned} \tag{3.8}$$

By Proposition 3.2 plus the fact that $a_{ij} = a_{ji}$ and $A_{xy} = A_{yx}$, it's easy to see that

$$\begin{aligned}
\mathrm{CD}(D) &= -\frac{2}{N}\sum_{x\in\Omega_s}\left(v_x + \frac{N}{s^2}\right)B_x + \frac{1}{N^2}\sum_{x\in\Omega_s}\sum_{y\in\Omega_s}\left(v_x + \frac{N}{s^2}\right)\left(v_y + \frac{N}{s^2}\right)A_{xy} + C_s \\
&= -\frac{2}{N}\sum_{x\in\Omega_s} v_x B_x + \frac{2}{N^2}\sum_{x\in\Omega_s}\sum_{y\in\Omega_s} v_x\left(\frac{N}{s^2}\right)A_{xy} + \frac{1}{N^2}\sum_{x\in\Omega_s}\sum_{y\in\Omega_s} v_x v_y A_{xy} + \text{constant.}
\end{aligned}$$

If s is odd, the first two terms in the last equation cancel each other out because of the first assertion in (3.8). Thus, we obtain

$$\mathrm{CD}(D) = \frac{1}{N^2}\sum_{x\in\Omega_s}\sum_{y\in\Omega_s} v_x v_y A_{xy} + \text{constant.}$$

If s is even, the equation above still holds, since $\sum_{x\in\Omega_s} v_x b_{x_1} = \sum_{x_1\in\mathbb{Z}_s}\sum_{x_2\in\mathbb{Z}_s} v_x b_{x_1} = 0$ and $\sum_{x\in\Omega_s} v_x b_{x_2} = \sum_{x_2\in\mathbb{Z}_s}\sum_{x_1\in\mathbb{Z}_s} v_x b_{x_2} = 0$, which hold because D is balanced and $\sum_{x_2\in\mathbb{Z}_s} v_x = \sum_{x_1\in\mathbb{Z}_s} v_x = 0$. The last step is to find the constant. We can simply take $v_x = 0$ for all x, or equivalently, $f_x = N/s^2$ for all x. Then, Proposition 3.1 says that the constant must be O_s, which proves the first equality of (3.6).

To prove the second equality, let $Q_s = \{(x_1, x_2) \in \Omega_s : x_1, x_2 \le s_1\}$. In (3.4), if we allow i and j to be any real number in $[0, s-1]$, then as a function on $[0, s-1] \times [0, s-1]$, A_{xy} is symmetric about $x = (s-1)/2$ and about $y = (s-1)/2$. Thus, it suffices to show the equations below.

(i) If s is odd, then

$$s^2 \sum_{x\in Q_s}\sum_{y\in Q_s} v_x v_y A_{xy} = \sum_{i=0}^{s_1}\sum_{j=0}^{s_1}\left(\sum_{x_1\le i, x_2\le j} v_x\right)^2.$$

(ii) If s is even, then

$$\begin{aligned} s^2 \sum_{x\in Q_s}\sum_{y\in Q_s} v_x v_y A_{xy} =& \frac{1}{4}\left(\sum_{x_1,x_2\le s_1} v_x\right)^2 + \frac{1}{2}\sum_{j=0}^{s_1-1}\left\{\left(\sum_{x_1\le j, x_2\le s_1} v_x\right)^2 + \left(\sum_{x_1\le s_1, x_2\le j} v_x\right)^2\right\} \\ &+ \sum_{i=0}^{s_1-1}\sum_{j=0}^{s_1-1}\left(\sum_{x_1\le i, x_2\le j} v_x\right)^2. \end{aligned}$$

We first consider (i). For $x = x^* = (x_1^*, x_2^*), y = y^* = (y_1^*, y_2^*) \in Q_s$, the term $v_{x^*}v_{y^*}$ will appear in the expansion of $(\sum_{x_1\le i, x_2\le j} v_x)^2$, if and only if $i \ge \max(x_1^*, y_1^*)$ and $j \ge \max(x_2^*, y_2^*)$. Thus, the coefficient of $v_{x^*}v_{y^*}$ on the right-hand-side is $2(s_1 - m_1 + 1)(s_1 - m_2 + 1)$, where $m_1 = \max(x_1^*, y_1^*)$ and $m_2 = \max(x_2^*, y_2^*)$. The proof can be completed if we can show that $2s^2 A_{x^*y^*} = 2(s_1 - m_1 + 1)(s_1 - m_2 + 1)$. By (3.4), it is easy to verify that $a_{ij} = a_{ji} = a_{ii}$ if $i \le j < (s-1)/2$, so we have $s^2 A_{x^*y^*} = s^2 a_{m_1 m_1} a_{m_2 m_2} = 4^{-1}\,|2m_1 + 1 - s|\,|2m_2 + 1 - s| = (s_1 - m_1 + 1)(s_1 - m_2 + 1)$, which proves (i). To prove (ii), the coefficient of $v_{x^*}v_{y^*}$ on the right-hand-side is $2\{4^{-1} + 2^{-1}(s_1 - m_1) + 2^{-1}(s_1 - m_2) + (s_1 - m_1)(s_1 - m_2)\} = 2(s_1 - m_1 + 0.5)(s_1 - m_2 + 0.5)$. Simple algebra shows that it's equal to $2s^2 A_{x^*y^*}$. □

We use two lemmas to provide some structure to the proof of Theorem 3.3.

Lemma 3.1. *Let F_s be a design that maximizes* $\mathrm{CD}(\cdot)$ *among all balanced s-level $s \times 2$ designs. Then,* $\mathrm{CD}(F_s) \geq \mathrm{CD}(D)$ *for any balanced s-level two-factor design D.*

Proof. Consider (3.5). Under a fixed N, $\mathrm{CD}(\cdot)$ can be seen as a linear function plus a quadratic function of f_x's. We denote this function by $\mathrm{CD}_N(f)$, where $f = \Big(f_{(0,0)}, \ldots, f_{(s-1,s-1)}\Big)$. We call f the frequency vector, which can fully characterize a design. Two facts are needed. First, $\mathrm{CD}_N(f)$ is a convex function of f, since Ma, Fang, and Lin [38] have shown that the quadratic part of $\mathrm{CD}_N(f)$ is positive-definite. Second, it is easy to verify that $\mathrm{CD}_N(f) = \mathrm{CD}_{kN}(kf)$ for any positive real numbers N, k, and $f \in \mathbb{R}^{s^2}$ satisfying $\sum_x f_x = N$. Let D an arbitrary s-level $N \times 2$ balanced with $N = rs$. Denote the frequency vectors of F_s and D by f_s and g, respectively. We prove by induction of r. When $r = 2$, there exist g' and g'' such that $g = g' + g''$ and both g' and g'' are balanced s-level $s \times 2$ designs. Thus,

$$\begin{aligned}\mathrm{CD}_s(f_s) &= \frac{1}{2}\mathrm{CD}_s(f_s) + \frac{1}{2}\mathrm{CD}_s(f_s) \geq \frac{1}{2}\mathrm{CD}_s(g') + \frac{1}{2}\mathrm{CD}_s(g'') \\ &\geq \mathrm{CD}_s\left(\frac{1}{2}g' + \frac{1}{2}g''\right) = \mathrm{CD}_s\left(\frac{1}{2}g\right) = \mathrm{CD}_{2s}(g).\end{aligned}$$

When $r = 3$, there exist an s-run design h' and $2s$-run design h'' such that, $g = h' + h''$ and h' and h'' are both balanced. Then,

$$\begin{aligned}\mathrm{CD}_s(f_s) &= \frac{1}{3}\mathrm{CD}_s(f_s) + \frac{2}{3}\mathrm{CD}_s(f_s) \geq \frac{1}{3}\mathrm{CD}_s(h') + \frac{2}{3}\mathrm{CD}_{2s}(h'') \\ &= \frac{1}{3}\mathrm{CD}_{2s}(2h') + \frac{2}{3}\mathrm{CD}_{2s}(h'') \geq \mathrm{CD}_{2s}\left(\frac{1}{3}(2h') + \frac{2}{3}(h'')\right) \\ &= \mathrm{CD}_{2s}\left(\frac{2}{3}g\right) = \mathrm{CD}_{3s}(g).\end{aligned}$$

For $r \geq 4$, the argument is similar. Therefore, we conclude that $\mathrm{CD}_s(f_s) \geq \mathrm{CD}_{rs}(g)$ for any balanced $(rs) \times 2$ design g, no matter $r = 1, 2, 3, \ldots$. □

Let D be an $\mathrm{SOA}_\alpha(N, 2, s, 2+)$ with index λ. Consider the $s \times s$ grid. Partition it into $(s')^2$ disjoint $\alpha \times \alpha$ sub-grids, and call a sub-grid an α-component of D. If treated as an α-level and two-factor design, an α-component must be balanced and contains $\lambda\alpha$ points.

Let G be an α-component of D. By Theorem 3.1, we can write $\mathrm{CD}(G) = O_\alpha + \alpha^{-4}\lambda^{-2}K(G)$, where $K(G) = \sum_x \sum_y v_x v_y A_{xy}$.

Lemma 3.2. *Let D be an $\mathrm{SOA}_\alpha(N, 2, s, 2+)$ and G_i be its ith α-component, $i = 1, 2, \ldots, (s')^2$. We have that $E(D) = (sN)^{-2}\{K(G_1) + \cdots + K(G_{(s')^2})\}$.*

Proof. We consider two special cases. First, let $s = 4$ and $\alpha = 2$. By Example 3.1 we see that $E(D) = (4N)^{-2}(v^2_{(0,0)} + v^2_{(0,3)} + v^2_{(3,0)} + v^2_{(3,3)})$. Since $v^2_{(0,0)} = v^2_{(0,1)} = v^2_{(1,0)} = v^2_{(1,1)}$ when D is an $\mathrm{SOA}_2(N, 2, 4, 2+)$, we have $v^2_{(0,0)} = 4^{-1}(v^2_{(0,0)} + v^2_{(0,1)} + v^2_{(1,0)} + v^2_{(1,1)}) = K(G_1)$. Similar argument can be made for G_2, G_3, and G_4, so we prove this special case.

Another case is given by $s = 9$ and $\alpha = 3$. If D is an $\mathrm{SOA}_3(N, 2, 9, 2+)$, similar argument and some simple algebra show that $V_{4\times 4} = K(G_5)$, $\sum_{k=1}^3 V_{k,4} + \sum_{l=1}^3 V_{4,l} = K(G_2) + K(G_4) + K(G_6) + K(G_8)$, and $\sum_{k=1}^3 \sum_{l=1}^3 V_{k\times l} = K(G_1) + K(G_3) + K(G_7) + K(G_9)$, which leads to $E(D) = (9N)^{-2}\{K(G_1) + \cdots + K(G_9)\}$.

The general proof uses the same idea but requires more complex notation and involves some tedious calculations, so we omit it. □

Proof of Theorem 3.3. We only need to prove $E(D) \leq k_\alpha/s^4$ since $\Psi(D) = 0$ is asserted by Theorem 3.2. Since $\phi(D)$ takes the average of $\mathrm{CD}(D_{ij})$, it suffices to prove this theorem for two-factor designs. Let D be an $\mathrm{SOA}_\alpha(N, 2, s, 2+)$. By Lemma 3.2, $E(D) = (sN)^{-2}\{K(G_1) + \cdots + K(G_{(s')^2})\}$. Lemma 3.1 says that $K(G_i) = (\mathrm{CD}(G_i) - O_\alpha)\alpha^4\lambda^2 \leq (\mathrm{CD}(F_\alpha) - O_\alpha)\alpha^4\lambda^2$. Simple algebra shows that $E(D) \leq k_\alpha/s^4$ with $k_\alpha = (\mathrm{CD}(F_\alpha) - O_\alpha)\alpha^4$. □

Proof of Theorem 3.4. Let $s = 2s'$ and $N = ss'$. It suffices to show that for any balanced s-level $N \times 2$ design D, $E(D) \geq 1/(4s^4)$. Consider (3.7). We know that $E(D) = (sN)^{-2}\sum q_{kl}V_{k\times l}$, where the sum is taken over all odd k and l, called odd terms for convenience. Since $(\sum_x v_x)^2 \geq 1/4$ if the sum is taken over an odd number of different x's, we have $E(D) \geq (sN)^{-2}\sum q_{kl}$. We first assume s' is even. In this case, among all $(s'/2)^2$ odd terms, $q_{kl} \equiv 1$, which implies $E(D) \geq (sN)^{-2}(s'/2)^2 = 1/(4s^4)$. If s' is odd, among all $\{(s'+1)/2\}^2$ odd terms, $q_{kl} = 1/4$ happens once, and $q_{kl} = 1/2$ happens $(s'-1)$ times, and $q_{kl} = 1$ happens $\{(s' - 1)/2\}^2$ times. Simple algebra leads to $E(D) \geq 1/(4s^4)$, and we complete the proof. □

Proof of Theorem 3.5. Let $s = 3s'$ and $N = ss'$. It suffices to show that for any balanced s-level $N \times 2$ design D, $E(D) \geq 4/(9s^4)$. Consider (3.7). We know that $E(D) = (sN)^{-2} \sum q_{kl} V_{k \times l}$, where the sum is taken over all k and l that are not multiples of 3. Since $(\sum_x v_x)^2 \geq 1/9$ if the sum is not taken over exactly $3k$ different x's, we have $E(D) \geq (sN)^{-2} \sum 4q_{kl}/9$. Like the proof of Theorem 3.4, a discussion based on whether s' is even or not and some simple algebra can show that $E(D) \geq 4/(9s^4)$. □

Chapter 4

A two-phase space-filling sub-data selection method for big data

4.1 Introduction

Modern statistics has been developed for over one century, but not until the last two decades have statisticians encountered the problem caused by big data volume. Big data is omnipresent nowadays because datum can be cheaply and routinely collected due to the rapid development of technologies and the popularity of the Internet. With a big dataset, even a basic analysis, such as the ordinary linear regression, may not be computationally affordable to everyone. As a result, data reduction becomes necessary.

We consider in this chapter a big dataset that has a very large number N of observations, while the number p of covariates is moderate. One solution to deal with big data volume is to conduct a statistical analysis using a small subset of the dataset. If the sub-dataset is well chosen, the statistical inferences based on it will be similar to those based on the full dataset. The most straightforward method is the simple random sampling (SRS); see Drineas, Mahoney, Muthukrishnan, and Sarlós [39], for example. Work on nonuniform sampling methods includes but is not limited to Drineas, Mahoney, and Muthukrishnan [40], Drineas, Mahoney, Muthukrishnan, and Sarlós [39], Drineas, Magdon-Ismail, Mahoney, and Woodruff [41], Ma, Mahoney, and Yu [15], and Ma and Sun [42], most of which focus on a leverage-based sampling, and/or a fast way to approximate the leverage scores.

Recently, motivated by the idea of optimal design of Kiefer [43], Wang, Yang, and Stufken [16] developed a deterministic method called information-based optimal sub-data selection

(IBOSS), which aims to select a sub-dataset that maximizes the information matrix. The sub-dataset obtained by the IBOSS method is similar to a D-optimal design, which consists of extreme points for better estimation under a linear model. Wang, Yang, and Stufken [16] showed that the variance of the IBOSS estimator converges to zero as the size of the full dataset N goes to infinity, even if the size of the sub-dataset is fixed. This feature is unavailable for a subsampling method.

Except for the SRS, all the aforementioned sub-data selection methods are model-dependent, with the main focus on the first-order linear model. However, the true model may actually contain some second-order terms or may not be even linear. Shi and Tang [44] utilized the idea of space-filling designs and proposed a model-independent method that selects a space-filling sub-dataset within the data region. They showed by simulations that a space-filling sub-dataset enjoys some robust properties.

In this chapter, we consider a non-parametric setting that the mean response is given by an unknown function $f(x)$. Intuitively, a space-filling sub-dataset can better capture the overall pattern of $f(x)$, because the selected points are uniformly distributed throughout the data region. Unlike Shi and Tang [44], who used the maximin criterion to select a space-filling sub-dataset, we present an alternative method that selects an orthogonal array-structured sub-dataset. More importantly, we propose a two-phase sub-data selection method that utilizes the information of response. In the existing methods, the response is ignored in the sub-data selection process. In fact, it is anything but clear how the response can play a role under a linear model or a parametric setting. In our case, since $f(x)$ may be smooth in some region but rugged in the other, it is reasonable to select more data points in the region where $f(x)$ is more rugged. Clearly, to identify the rugged areas, the information of response must be taken into account.

The remainder of this section introduces some notation. Consider a dataset consisting of N data points $(x_1, y_1), \ldots, (x_N, y_N)$, where $x_i = (x_{i1}, \ldots, x_{ip})$ is a p-dimensional covariate vector corresponding to the response variable y_i. The number p of covariates is assumed moderate and $N \gg p$. We let $E(y_i|x_i) = f(x_i)$ $(i = 1, \ldots, N)$, where $f(x)$ is unknown. We want to select a sub-dataset containing n points $(x_{k_1}, y_{k_1}), \ldots, (x_{k_n}, y_{k_n})$ and then construct a nonparametric estimate of $f(x)$ using the selected sub-dataset. In Section 4.2, we present an algorithm for selecting a sub-dataset that has an orthogonal array (OA) structure, thus forming a space-filling subset in the data region. In Section 4.3, we develop a two-phase sub-data selection method that uses our OA-based algorithm in the first phase, and then selects additional data points in the regions where $f(x)$ is rugged in the second phase. Simulation studies are given in Section 4.4.

4.2 An OA-based sub-data selection method

The idea of selecting a space-filling sub-dataset has been used in Shi and Tang [44], who selected a sub-dataset that maximizes the minimum distance between data points. As shown in Shi and Tang [44], a space-filling sub-dataset is advantageous. For example, when a first-order linear model is fitted but the true model actually contains some second-order terms, their method has a smaller prediction mean squared error than the IBOSS method.

We present a space-filling sub-data selection method based on OAs. Recall that an s-level OA of strength t, denoted by $\mathrm{OA}(n, p, s, t)$, is an $n \times p$ matrix in which each column has s levels from $\mathbb{Z}_s = \{0, 1, \ldots, s-1\}$ such that, for any t columns, all possible level-combinations appear equally often. As discussed in Chapter 3, OAs are one kind of space-filling designs that achieves uniformity in low dimensions, which is important and useful in many situations. For example, under an additive model $E(y_i|x_i) = \beta_0 + f_1(x_{i1}) + \cdots + f_p(x_{ip})$, if a sub-dataset forms an OA of strength one, it can better capture each $f_j(\cdot)$ because of its uniformity in every one dimension.

To obtain an OA-structured sub-dataset of size n, we start with an OA of size $n \times p$, denoted by $M = [m_{ij}]$ ($m_{ij} \in \mathbb{Z}_s$). We next do some preprocessing on the full dataset $X = (x_1^T, \ldots, x_N^T)^T$. Let $R(x_{ij}) = 1$ if x_{ij} is the smallest element of the jth column of X, $R(x_{ij}) = 2$ if x_{ij} is the second smallest element of the jth column of X, and so on. If there are ties, say for example, both x_{1j} and x_{2j} are smallest elements in the jth column of X, then they will be randomly ranked as if they are distinct. Hence, for a fixed j, $R(x_{ij})$'s must be a permutation of $1, \ldots, N$. Assume N is a multiple of s and $N = rs$. We construct a matrix $X^* = [x_{ij}^*]$ by letting $x_{ij}^* = \lfloor (R(x_{ij}) - 1)/r \rfloor$, where $\lfloor \cdot \rfloor$ stands for the integer part of a number. Clearly, X^* is a balanced design matrix of s levels, which means that within each column of X^*, each element of $\mathbb{Z}_s$ appears equally often. If N is not a multiple of s, we can still take $r = N/s$ and $x^*_{ij} = \lfloor (R(x_{ij}) - 1)/r \rfloor$, and the resulting X^* is nearly balanced. We use x^*_k to denote the kth row of X^* and m_i the ith row of M. Our OA-based method is given as follows.

Algorithm 4.1. *Start with $S = \phi$ and $i = 1$.*

1. *Among $\{x_1, \dots, x_N\} \setminus S$, randomly select one x_k, denoted by x_{k_i}, among those x_k's with x^*_k having the smallest L_1-distance to m_i, where the L_1-distance is $d_1(x^*_k, m_i) = \sum_{j=1}^{p} |x^*_{kj} - m_{ij}|$.*
2. *Set $S = S \cup \{x_{k_i}\}$. If $i = n$, the algorithm stops and $S = \{x_{k_1}, \dots, x_{k_n}\}$ is the subdataset we obtain. Otherwise, go back to step 1 with $i = i + 1$.*

One can use any distance other than the L_1-distance in step 1. To obtain x_{k_i}, we need to compute $(N - i + 1)n$ distances ($i = 1, \dots, n$), so the complexity of Algorithm 4.1 is $O(Nn^2)$, which is essentially $O(N)$ as $N \gg n$.

A similar algorithm that doesn't need to construct X^* is as follows. First, rescale X and M into a unit hypercube $[0, 1]^p$. Then, for $i = 1, \dots, n$, find the x_k, denoted by x_{k_i}, that has the smallest $d_1(x_k, m_i)$ value. This alternative algorithm may be more natural than Algorithm 4.1, but it actually ignores the distribution of the data points. To see this, suppose that $p = 1$ and M is $[0, 1, 2]^T$, a one-column and three-level OA. Then, Algorithm 4.1 selects as x_{k_2} an x_k from those in the mid-third group given by $x_{(i)}$ with $N/3 < i \leq 2N/3$, while the alternative algorithm selects an x_k that is closest to $\max(x_k)/2$. These two quantities can be very different if the distribution of $x_1, \dots, x_N$ is highly skewed.

In Algorithm 4.1, if one uses an M that is an OA of strength t, then the obtained subdataset is space-filling in every d-dimension with $d \leq t$. Thus, an M of a higher strength is preferred. But using such an M is expensive, so we will focus on using an $\mathrm{OA}(n, m, s, t)$ with a small t and large s. For instance, in Example 4.1, an $\mathrm{OA}(16, 5, 4, 2)$ is considered, and in Section 4.4.2, we use an $\mathrm{OA}(900, 2, 30, 2)$ in the simulation.

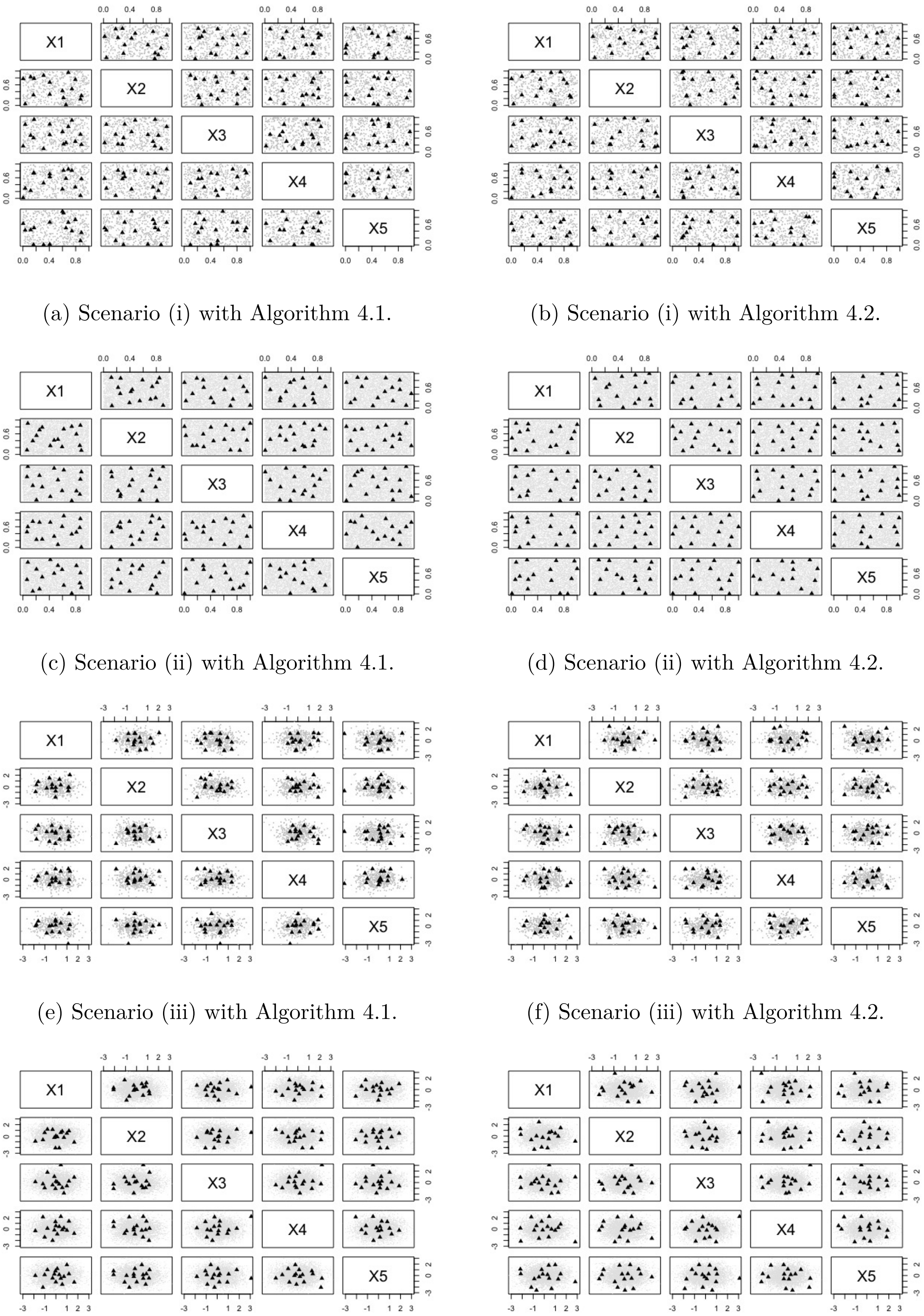
In step 1 of Algorithm 4.1, as there are many x_k with x_k^* having the smallest L_1-distance to m_i, we can do a further optimization to make the sub-dataset more space-filling. First, for each column of X, we find s equally-spaced percentiles. For example, if $s = 3$, find the 0th, 50th, and 100th percentiles; if $s = 4$, find the 0th, 33th, 67th, and the 100th percentiles, where the 0th and 100th percentiles are the minimum and maximum, respectively. Let $q_{j,0}, \ldots, q_{j,s-1}$ denote such s percentiles of the jth column of X. Now, for an x_k, we define $\xi(x_k) = \sum_{j=1}^{p} |x_{kj} - q_{j,m_{ij}}|$. When several x_k's can be chosen in step 1, we choose the one that has the smallest $\xi(x_k)$ value. For example, if $m_1 = (0, \ldots, 0)$ and there are several x_k with $d_1(x_k^*, m_1) = 0$. Among these x_k's, the smaller the value of $\xi(x_k) = \sum_{j=1}^{p} |x_{ij} - q_{j,0}|$ is, the closer an x_k is to the point $(q_{1,0}, \ldots, q_{p,0})$, and the one that has the smallest $\xi(x_k)$ value can best represent m_1. The next algorithm summarizes this modification.

Algorithm 4.2. *Obtain a sub-dataset by Algorithm 4.1, but replace step 1 by step 1' below.*

1'. *Among $\{x_1, \ldots, x_N\} \setminus S$, consider those x_k's with x_k^* having the smallest $d_1(x_k^*, m_i)$. Among these x_k's, choose the one that has the smallest $\xi(x_k)$ value, and denote this x_k by x_{k_i}.*

We end this section by an illustrative example.

Example 4.1. *We generate $x_1, \ldots, x_N$ with $p = 5$ under four different scenarios: (i) $N = 500$ and the x_i's are independent and have a multivariate uniform distribution on the unit cube $(0, 1)^p$, (ii) x_i's are generated in the same way as in (i) but with $N = 5000$, (iii) $N = 500$ and the x_i's are independent and have a standard multivariate normal distribution, and (iv) the same as in (iii) but with $N = 5000$. We take M to be an* $\mathrm{OA}(16, 5, 4, 2)$*, and use Algorithm 4.1 to select a sub-dataset of size $n = 16$. The pairwise scatter plots under each scenario are given in the left columns of Figure 4.1. In these plots, the gray dots are all the data points and the black triangles are the selected ones. The results from using Algorithm 4.2 are given in the right columns of Figure 4.1. The difference between Algorithms 4.1 and 4.2 is most noticeable in Figures 4.1c and 4.1d.*



(a) Scenario (i) with Algorithm 4.1.

(b) Scenario (i) with Algorithm 4.2.

(c) Scenario (ii) with Algorithm 4.1.

(d) Scenario (ii) with Algorithm 4.2.

(e) Scenario (iii) with Algorithm 4.1.

(f) Scenario (iii) with Algorithm 4.2.

(g) Scenario (iv) with Algorithm 4.1.

(h) Scenario (iv) with Algorithm 4.2.

Figure 4.1: The pairwise scatter plots under each scenario.

4.3 A two-phase method using response information

In this section, we develop a sub-data selection method that utilizes the information on y_i. Existing methods such as a leverage-based subsampling or the IBOSS method mainly focus on the estimation under a linear model and do not allow y_i to play a role in the process. The basic idea of our method is as follows. In the area where $f(x)$ has a rugged surface, more data points should be selected. In contrast, in the area where $f(x)$ is smooth, we should select less points. To be more specific, our method contains two phases. In the first phase, we select n_1 points that spread uniformly throughout the data region. Among these n_1 points, we identify those points at which $f(x)$ has large curvature, called rugged points for convenience. Then, we uniformly select another n_2 points from the neighbor points of the rugged points in the second phase.

The way we identify the rugged points is as follows. For a point x_i selected in the first phase, find all the points in its δ-neighborhood. Use x_i and all its neighbor points to fit a first-order linear model, and then find the residuals. We use the average of the squared residuals, denoted by R_i, to measure the degree of $f(x)$ deviating from a plane in a local area around x_i. If R_i is large, then $f(x)$ is rugged around x_i.

We summarize our method in an algorithm below.

Algorithm 4.3. *We obtain a sub-dataset as follows.*

1. *Among all data points, select n_1 points using Algorithm 4.2.*

2. *For each x_i selected in step 1, fit a first-order linear model within its δ-neighborhood. Find the average of the squared residuals and denote it by R_i.*

3. *Identify the x_i's having the n^* largest R_i values. Without loss of generality, let these points be $x_1, \ldots, x_{n^*}$*

4. *Let U collect all points, minus the n_1 points in step 1, in the δ-neighborhoods of $x_1, \ldots, x_{n^*}$. Using Algorithm 4.2, select n_2 points out of U. These n_2 points together with the n_1 points then form a sub-dataset containing $n = n_1 + n_2$ points.*

Remark 4.1. In steps 1 and 4, Algorithm 4.2 can be replaced by Algorithm 4.1, simple random sampling (SRS), or other methods.

The δ-neighborhood can be defined using any distance, and we use the symmetric nearest neighborhood, as introduced below. We first consider $p = 1$. In this case, $x_1, \dots, x_N$ are all real numbers. Let $R(x_i)$ be the rank of x_i within $x_1, \dots, x_N$. For a point x_i, we define its symmetric nearest neighborhood of width ω as $B(x_i, \omega) = \{x_k : |R(x_k) - R(x_i)| \leq \omega\}$. If $p \geq 2$, we deal with one dimension at a time, and obtain a neighborhood for each dimension. Then $B(x_i, \omega)$ is defined as the intersection of all these neighborhoods.

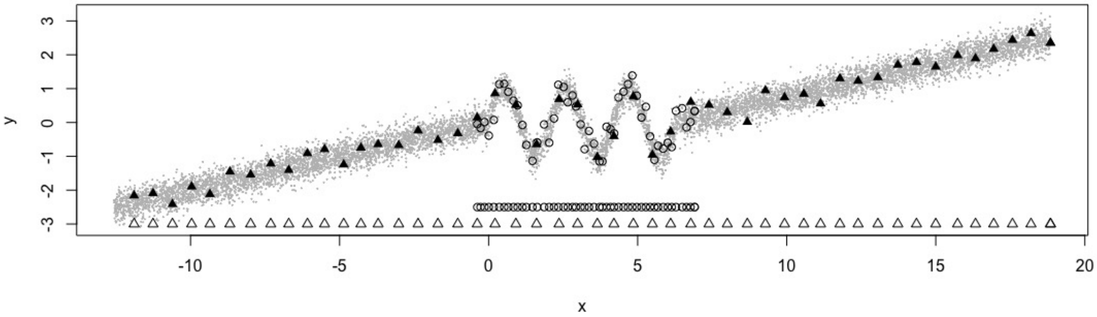
We illustrate Algorithm 4.3 by an example.

Example 4.2. *Consider a function $f(x)$ defined on $(-4\pi, 6\pi)$, where*

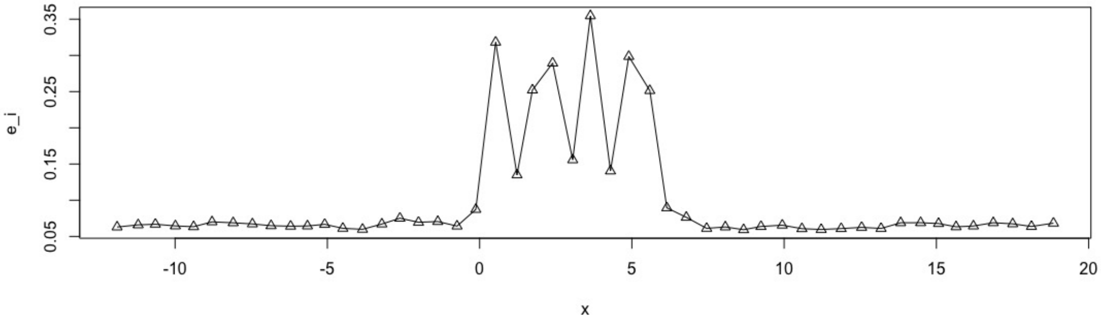
$$f(x) = \begin{cases} 0.2x & \text{if } x \in (-4\pi, 0]; \\ \sin(3x) & \text{if } x \in (0, 2\pi]; \\ 0.2(x - 2\pi) & \text{if } x \in (2\pi, 6\pi). \end{cases}$$

We generate a dataset of size $N = 10,000$ as follows: $x_1, \dots, x_N$ are i.i.d. uniform $U(-4\pi, 6\pi)$, and $y_i = f(x_i) + \epsilon_i$, where ϵ_i are i.i.d. normal $N(\mu = 0, \sigma^2 = 0.25^2)$. To implement Algorithm 4.3, in step 1, we set $n_1 = 50$ and use Algorithm 4.2 with M being $[0, \dots, 49]^T$, an one-column OA. *In step 2, the δ-neighborhood is $B(x_i, 250)$. In step 3, we set $n^* = 10$. In step 4, the δ-neighborhood is again $B(x_i, 250)$, and the $n_2 = 50$ points are similarly selected as in step 1.*

In Figure 4.2a, the gray dots stand for all the data points, black triangles for the n_1 points selected in step 1, and circles for the n_2 points selected in step 4. The projection points of the circles and triangles are also drawn. In Figure 4.2b, we can see the R_i values of the triangle points, which indicate that the $n^* = 10$ points in step 3 must fall in the interval $(0, 2\pi)$, and thus U approximately consists of all the points within $(0, 2\pi)$.



(a) Scatter plots of all/selected points and their projection.



(b) The R_i values of the $n_1 = 50$ triangle points.

Figure 4.2: Scatter plot of the data points and the R_i values.

4.4 Simulation studies

4.4.1 A one-dimensional case

In this subsection, a simulation with $p = 1$ is carried out. To fit the unknown function $f(x)$, we consider Gaussian kernel estimation. Without loss of generality, let $S = \{x_1, \ldots, x_n\}$ be the selected sub-dataset. Under S, the fitted value of $f(x)$ is given by

$$\hat{f}(x) = \frac{\sum_{i=1}^{n} g_h(x_i - x) y_i}{\sum_{i=1}^{n} g_h(x_i - x)}, \tag{4.1}$$

where $g_h(x)$ is the density function of a normal distribution $N(0, h^2)$. In our simulation, the bandwidth h is determined by the function "npreg" of an R package called "np", with default settings. This package also provides several different kernel smoothing methods, but we will only illustrate Gaussian kernel estimation.

Consider the $f(x)$ and dataset given in Example 4.2. We compare four different sub-data selection methods with sub-data size $n = 100$: (i) the SRS, (ii) Algorithm 4.2 with M being $[0, \ldots, 99]^T$, an one-column OA, (iii) the method used in Example 4.2, and (iv) the same as (iii) except for that in steps 1 and 4, Algorithm 4.2 is replaced by SRS. Note that methods (ii) and (iii) are deterministic. Let S_2 and S_3 be the sub-dataset obtained by (ii) and (iii), respectively. We repeat the following procedure 500 times.

1. Use methods (i) and (iv) to select sub-datum S_1 and S_4, respectively.

2. Use S_1, S_2, S_3, and S_4 to fit $f(x)$ by equation (4.1).

3. For each method, calculate the sum of the squared errors $SSE = \sum_{i=1}^{N}(\hat{f}(x_i) - f(x_i))^2$.

4. For a more detailed comparison, we also calculate $SSE^{sin} = \sum_{x_i \in (0,2\pi)}(\hat{f}(x_i) - f(x_i))^2$ and $SSE^{line} = \sum_{x_i \notin (0,2\pi)}(\hat{f}(x_i) - f(x_i))^2$.

5. Re-generate $\epsilon_1, \ldots, \epsilon_N$, which are i.i.d. $N(0, 0.25^2)$.

We use $\hat{f}_i(x)$ to denote the estimate of $f(x)$ based on S_i, $i = 1, 2, 3, 4,$. Figure 4.3 presents the results from the first repetition, where $f(x)$ is also drawn by a dotted line. The box plots of SSE, SSE^{sin}, and SSE^{line} from the four methods are given in Figures 4.4a, 4.4b, and 4.4c, respectively.

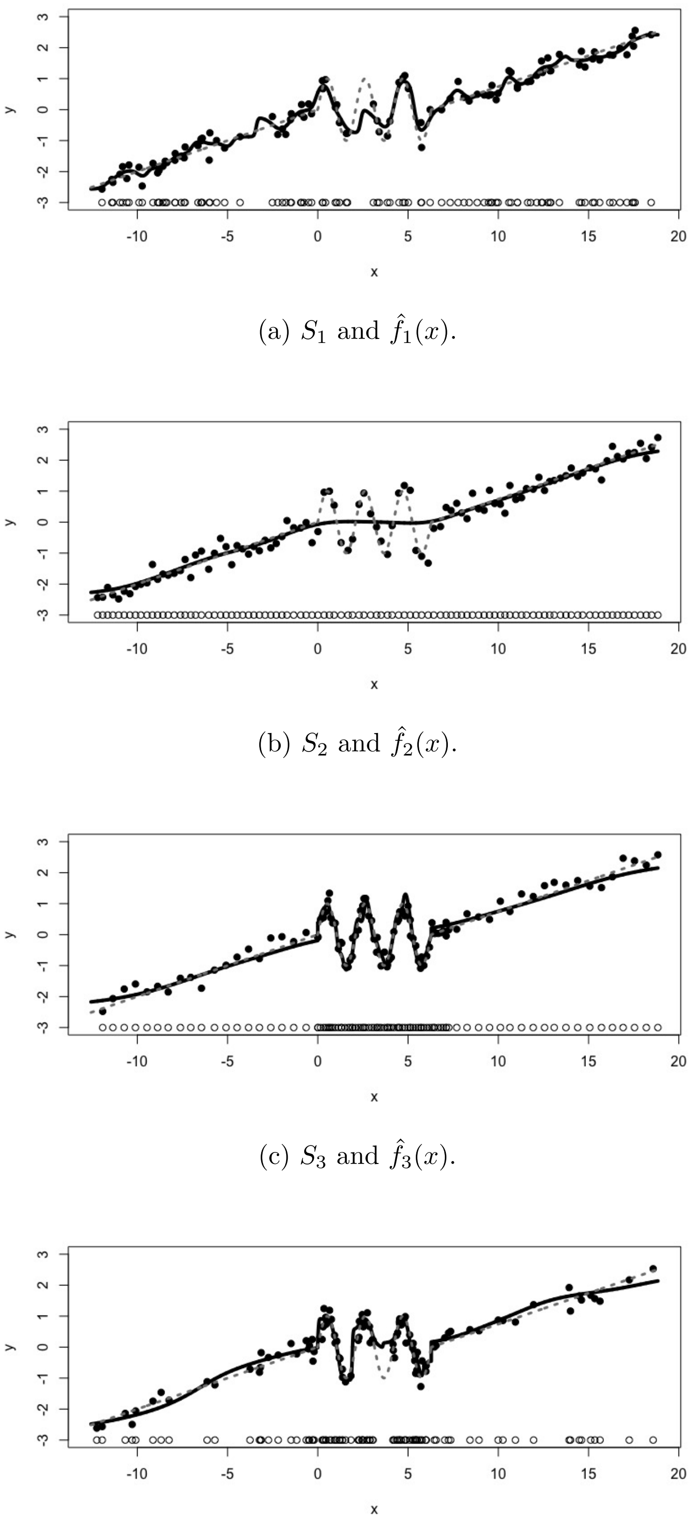
Recall that U is the union of the δ-neighborhoods of the n_1 points selected in the first phase. For methods (iii) and (iv), we make a small modification when computing $\hat{f}(x)$. If $x \notin U$, $\hat{f}(x)$ is obtained using only the points outside U. Otherwise, $\hat{f}(x)$ is obtained by equation (4.1) using all 100 points. The basic idea here is to allow two bandwidths, one for U and the other for U^C, since the R package "np" uses the same bandwidth for the entire range.

Some comments on the results in Figures 4.3 and 4.4 are given below.

A space-filling sub-dataset can lead to a better fitting of $f(x)$, as can be seen by noting that method (ii) is better than (i) in terms of SSE^{line} and that (iii) is better than (iv) in terms of SSE, SSE^{sin}, and SSE^{line}. Worth noting is that both methods (ii) and (iii) can avoid extremely large SSE, SSE^{sin}, or SSE^{line}, which happens in methods (i) and (iv).

Selecting more points in the area where $f(x)$ is rugged can improve the overall fitting, since in terms of SSE, method (iii) is better than (ii), and (iv) is better than (i). The success of method (iii) and (iv) is due to their much smaller SSE^{sin} values, at a price of slightly larger SSE^{line}.

Selecting more points in the rugged area is more important than selecting a space-filling sub-dataset. Note that S_2 consists of data points that are uniformly distributed throughout the entire data region, but method (ii) is the worst in terms of SSE. The reason is that $\hat{f}_2(x)$ almost always misses the sine function in $(0, 2\pi)$, so its SSE mostly comes from the bias of $\hat{f}_2(x_i)$ within $(0, 2\pi)$.



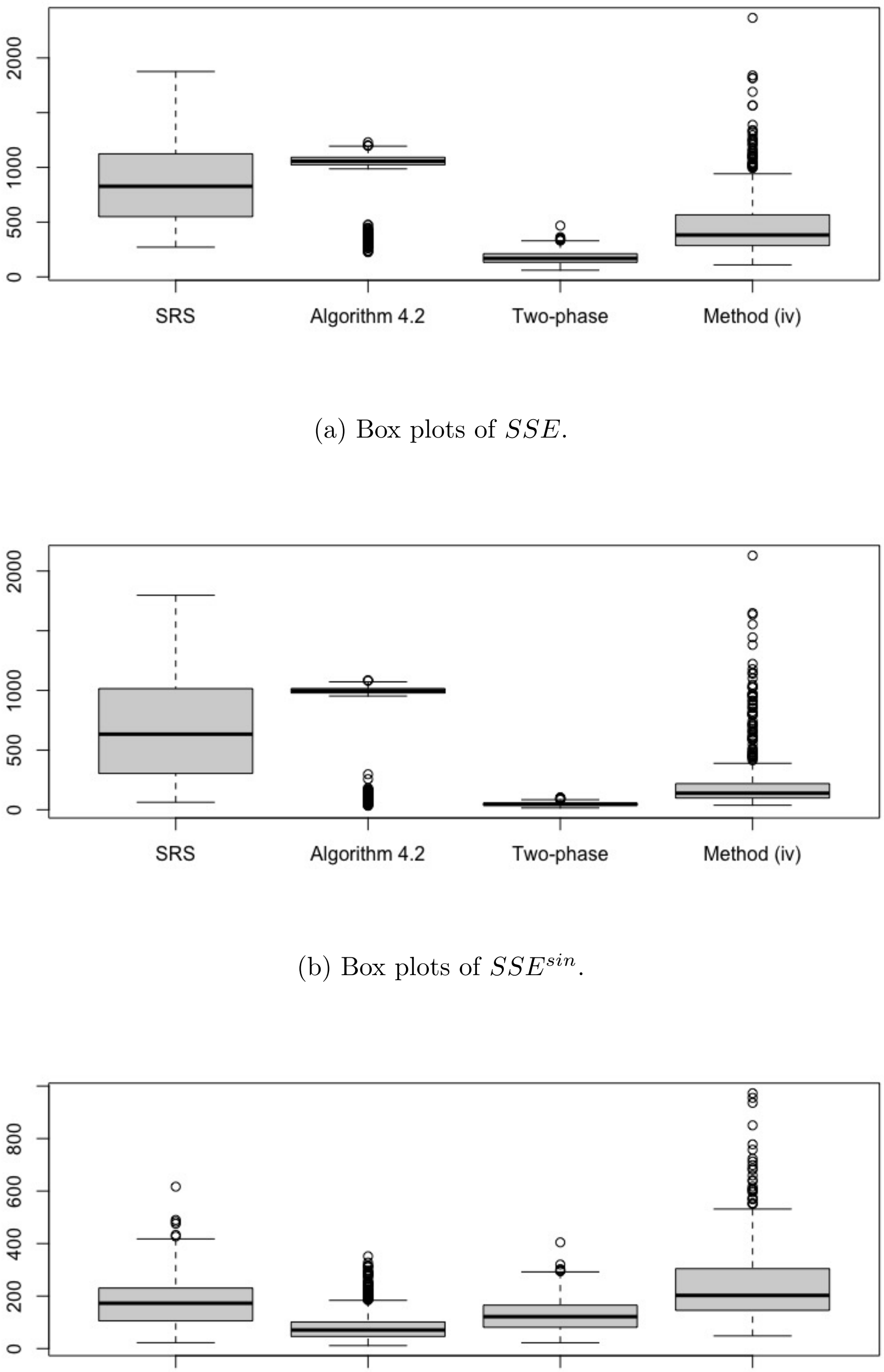
(a) S_1 and $\hat{f}_1(x)$.

(b) S_2 and $\hat{f}_2(x)$.

(c) S_3 and $\hat{f}_3(x)$.

(d) S_4 and $\hat{f}_4(x)$.

Figure 4.3: The scatter plots and fitted the lines under methods (i)-(iv).



(a) Box plots of SSE.

(b) Box plots of SSE^{sin}.

(c) Box plots of SSE^{line}.

Figure 4.4: The box plots of the sum of squared errors (500 times repetition).

4.4.2 A two-dimensional case

In this subsection, a simulation with $p = 2$ is carried out, and the notation defined in the previous subsection will be used similarly. We still consider the Gaussian kernel estimation. Without loss of generality, let $S = \{x_1, \ldots, x_n\}$ be the selected sub-dataset, where $x_i = (x_{i1}, x_{i2})$. As before, let $E(y_i|x_i) = f(x_i)$. The fitted value of $f(x)$ at a point $t = (t_1, t_2)$ is

$$\hat{f}(t) = \frac{\sum_{i=1}^{n} g_{h_1}(x_{i1} - t_1) g_{h_2}(x_{i2} - t_2) y_i}{\sum_{i=1}^{n} g_{h_1}(x_{i1} - t_1) g_{h_2}(x_{i2} - t_2)}. \tag{4.2}$$

In our simulation, the bandwidth (h_1, h_2) is determined by the R package "np".

Let $\mu_1 = (0, 1)$, $\mu_2 = (0, -1)$, $\mu_3 = (1, 0)$, $\mu_4 = (-1, 0)$, and let I_p denote the identity matrix of order p. Consider a function $f(t)$ defined as

$$f(t) = f(t_1, t_2) = -0.5(t_1 + t_2) + 3\left(G(t; \mu_1) + G(t; \mu_2) + G(t; \mu_3) + G(t; \mu_4)\right),$$

where $G(t; \mu)$ is the density of a bivariate normal distribution $N(\mu, 0.5I_2)$. We generate a dataset of size $N = 10,000$ as follows: $x_1, \ldots, x_N$ are i.i.d. bivariate uniform $U(-3, 3)^2$, and $y_i = f(x_i) + \epsilon_i$, where ϵ_i are i.i.d. normal $N(0, 0.2^2)$. We compare four different sub-data selection methods with $n = 900$, where the first three are as follows: (i) the SRS, (ii) Algorithm 4.2 with M being an OA$(900, 2, 30, 2)$, and (iii) Algorithm 4.3 with details given as follows. In step 1, we use Algorithm 4.2 with M being an OA$(576, 2, 24, 2)$, and thus $n_1 = 576$. In step 2, the δ-neighborhood is $B(x_i, 2000)$. In step 3, we set $n^* = 200$. In step 4, the δ-neighborhood is $B(x_i, 500)$, and the $n_2 = 324$ points are similarly selected as in step 1 with M being an OA$(324, 2, 18, 2)$. Method (iv) is similar to (iii) except for that in steps 1 and 4, Algorithm 4.2 is replaced by the SRS. We repeat the following procedure 500 times.

1. Use methods (i) and (iv) to select sub-datum S_1 and S_4, respectively.

2. Use S_1, S_2, S_3 and S_4 to fit $f(x)$ by equation (4.2).

3. For each method, calculate $SSE = \sum_{i=1}^{N}(\hat{f}(x_i) - f(x_i))^2$.

4. Calculate $SSE^{hill} = \sum_{x_i \in \mathcal{H}}(\hat{f}(x_i) - f(x_i))^2$ and $SSE^{flat} = \sum_{x_i \notin \mathcal{H}}(\hat{f}(x_i) - f(x_i))^2$, where the set $\mathcal{H}$ is defined in the next paragraph.

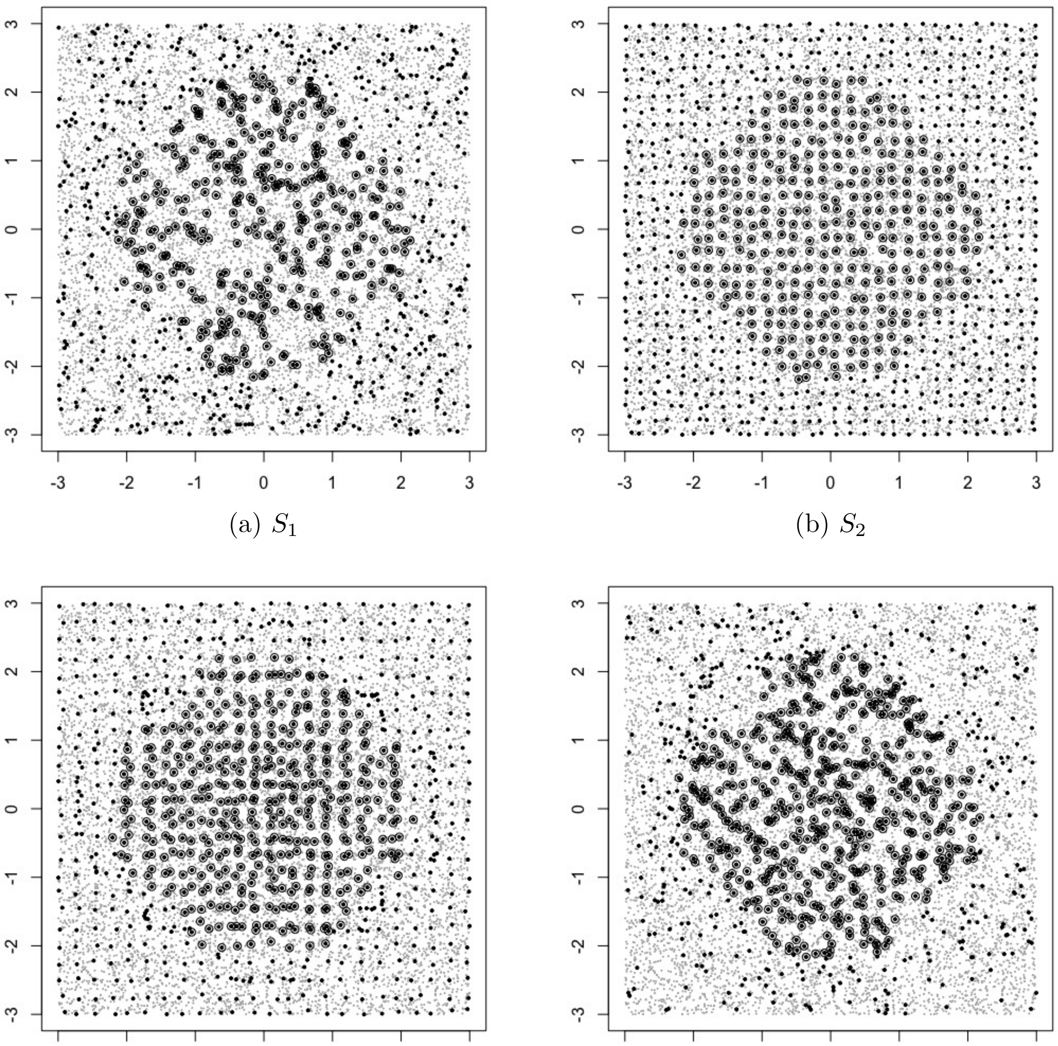
5. Re-generate $\epsilon_1, \ldots, \epsilon_N$, which are i.i.d. $N(0, 0.2^2)$.

For each x_i $(i = 1, \ldots, N)$, we find the R_i defined in step 2 of Algorithm 4.3 with the δ-neighborhood being $B(x_i, 500)$, which is calculated under the situation $\epsilon_i = 0$ for all i. Rank $x_1, \ldots, x_N$ by their R_i values in descending order. The first 4000 x_i's belong to $\mathcal{H}$, and the last 6000 to its complement.

We plot $S_1, \ldots, S_4$ in Figure 4.5 and the corresponding $\hat{f}_1, \ldots, \hat{f}_4$ in Figure 4.6. The surface of $f(x)$ is drawn in Figure 4.7. In Figure 4.5, the gray dots are all the data points, and black dots are the selected ones. A black dot is circled if it is in $\mathcal{H}$. The box plots of SSE, SSE^{hill}, and SSE^{flat} are given in Figures 4.8a, 4.8b, and 4.8c, respectively.

Let U be the union of the symmetric nearest neighborhoods of width 1000 of the n_1 points. Similar to the previous subsection, for methods (iii) and (iv), $\hat{f}(x)$ is obtained using only the points outside U if $x \notin U$.

Both methods (ii) and (iii) are far better than method (i) and (iv). Method (ii) performs the best in terms of SSE^{flat}, and method (iii) is the best in terms of SSE and SSE^{hill}. Method (iii) does not show a clear advantage over method (ii). This is likely because our choice of a relatively large n gives enough points in the hilly region even for method (ii).



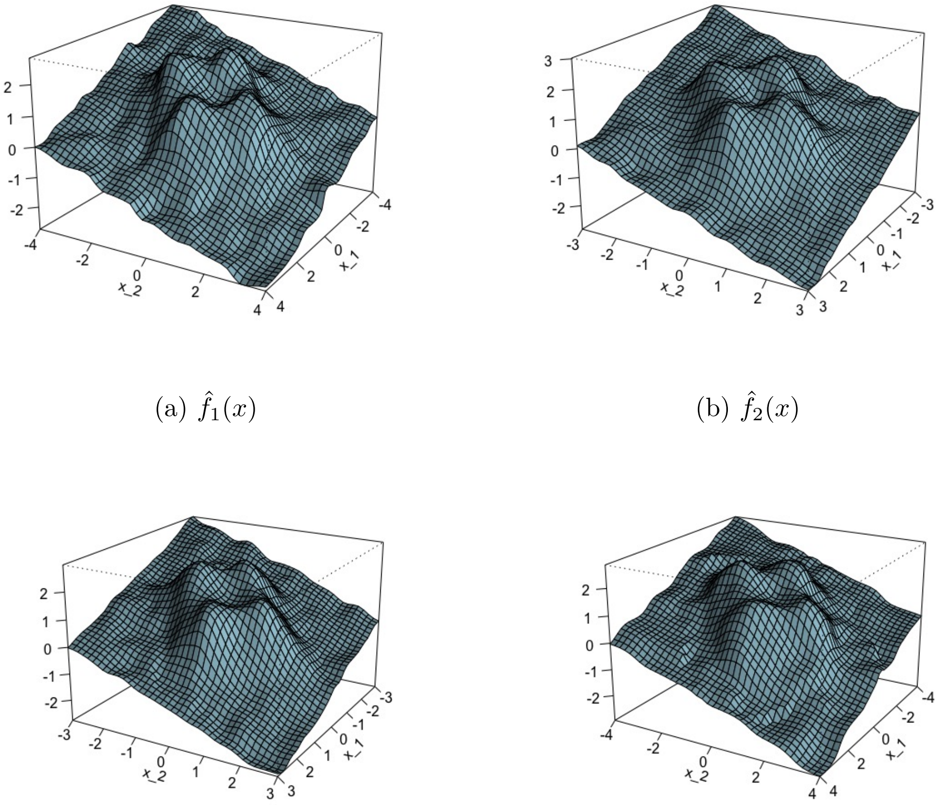
(a) S_1

(b) S_2

(c) S_3

(d) S_4

Figure 4.5: Scatter plots of S_1-S_4.



(a) $\hat{f}_1(x)$

(b) $\hat{f}_2(x)$

(c) $\hat{f}_3(x)$

(d) $\hat{f}_4(x)$

Figure 4.6: Fitted surfaces under S_1-S_4.

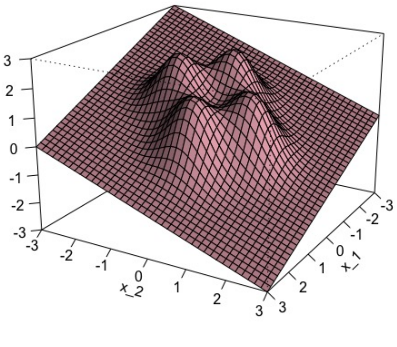
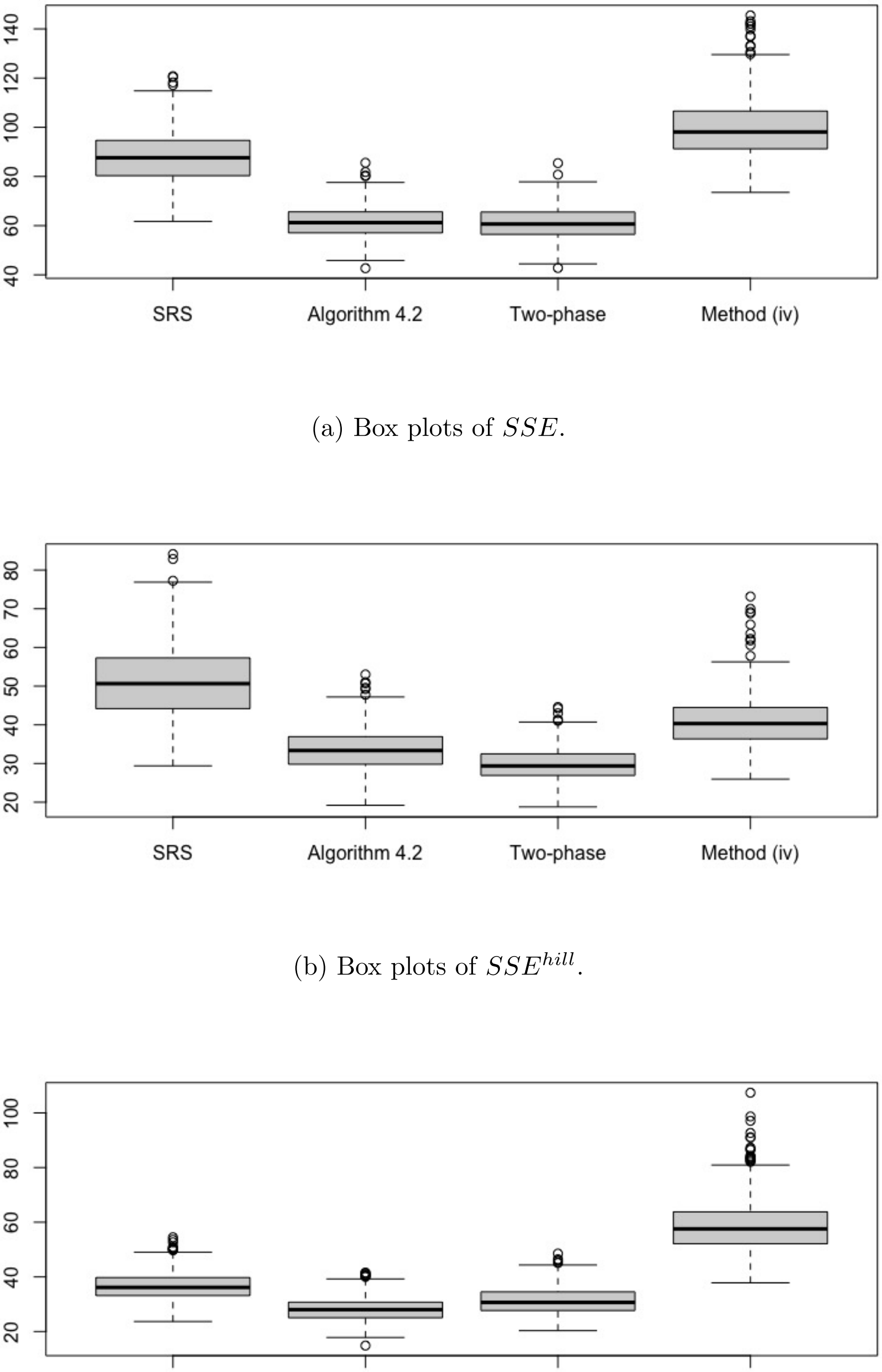


Figure 4.7: The surface of $f(x)$.



(a) Box plots of SSE.

(b) Box plots of SSE^{hill}.

(c) Box plots of SSE^{flat}.

Figure 4.8: The box plots of the sum of squared errors (500 times repetition).

Chapter 5

Summary and future work

This dissertation investigates three research problems. The first problem centers on the relationship between the OP and BP. We derive a linear relation between the two sets of factorial effects, and based on it obtain some useful results on design construction. The second one is to examine the performance of SOAs under the uniform projection criterion. We find a new and insightful expression of the centered L_2-discrepancy, and show that SOAs are optimal or nearly optimal uniform projection designs. The third is on the sub-data selection for big data. We present a sub-data selection method in which the sub-data points form a space-filling subset and more data points are selected in the area where the response surface is rugged. Simulations show that our method leads to a smaller prediction mean square error, as compared to the simple random sampling.

One possible future work from Chapter 2 is to build a complete catalog of OAs of small run sizes that includes all possible arrays under the baseline isomorphism [see 4]. This catalog is useful for finding optimal OAs under the BP, no matter what criterion or model one considers. The catalog is also useful for theoretical investigations as it allows easy testing of a potential theoretical result one is attempting to formulate.

Chapter 3 focuses on space-filling designs that enjoy good two-dimensional projection properties. It is natural to ask whether or not similar results can be obtained if we consider three or higher dimensional projections. The uniform projection criterion as in (3.3) can be easily generalized to three dimensions. We expect that the SOAs of strength three constructed by Shi and Tang [45] will perform well under this criterion.

The work of Chapter 4 will need more comprehensive simulations that consider more settings and perhaps also higher dimensional cases. It will need a real-data example as well. It would be also interesting to compare our OA-based sub-data selection method with the IBOSS method of Wang, Yang, and Stufken [16] and the maximin method of Shi and Tang [44] under the first and second order models.

Bibliography

[1] Yee Hwa Yang and Terry Speed. Design issues for cdna microarray experiments. *Nature Reviews Genetics*, 3(8):579–588, 2002.

[2] Kathleen F Kerr. Efficient 2^k factorial designs for blocks of size 2 with microarray applications. *Journal of Quality Technology*, 38(4):309–318, 2006.

[3] Tathagata Banerjee and Rahul Mukerjee. Optimal factorial designs for cdna microarray experiments. *Annals of Applied Statistics*, 2(1):366–385, 2008.

[4] Rahul Mukerjee and Boxin Tang. Optimal fractions of two-level factorials under a baseline parameterization. *Biometrika*, 99(1):71–84, 2012.

[5] Sigeiti Moriguti. Optimality of orthogonal designs. *Rep. Statist. Appl. Res. Un. Japan Sci. Engs*, 3:75–97, 1954.

[6] Cheng-Yu Sun and Boxin Tang. Relationship between orthogonal and baseline parameterizations and its applications to design constructions. *Statistica Sinica*, 32(1), 2022.

[7] Anqi Chen, Cheng-Yu Sun, and Boxin Tang. Selecting baseline designs using a minimum aberration criterion when some two-factor interactions are important. *Statistical Theory and Related Fields*, 5(2):1–7, 2021.

[8] Mark E Johnson, Leslie M Moore, and Donald Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131–148, 1990.

[9] Fred Hickernell. A generalized discrepancy and quadrature error bound. *Mathematics of Computation*, 67(221):299–322, 1998.

[10] Hyejung Moon, Angela Dean, and Thomas Santner. Algorithms for generating maximin latin hypercube and orthogonal designs. *Journal of Statistical Theory and Practice*, 5 (1):81–98, 2011.

[11] V Roshan Joseph, Evren Gul, and Shan Ba. Maximum projection designs for computer experiments. *Biometrika*, 102(2):371–380, 2015.

[12] Michael D McKay, Richard J Beckman, and William J Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61, 2000.

[13] Fasheng Sun, Yaping Wang, and Hongquan Xu. Uniform projection designs. *Annals of Statistics*, 47(1):641–661, 2019.

[14] Yuanzhen He, Ching-Shui Cheng, and Boxin Tang. Strong orthogonal arrays of strength two plus. *Annals of Statistics*, 46(2):457–468, 2018.

[15] Ping Ma, Michael Mahoney, and Bin Yu. A statistical perspective on algorithmic leveraging. *The Journal of Machine Learning Research*, 16:861–911, 2015.

[16] HaiYing Wang, Min Yang, and John Stufken. Information-based optimal subdata selection for big data linear regression. *Journal of the American Statistical Association*, 114(525):393–405, 2019.

[17] George EP Box and J Stuart Hunter. The 2^{k-p} fractional factorial designs. *Technometrics*, 3(3):311–351, 1961.

[18] CF Jeff Wu and Michael S Hamada. *Experiments: Planning, Analysis, and Optimization.* Hoboken, New Jersey: Wiley, 2nd edition, 2009.

[19] Pei Li, Arden Miller, and Boxin Tang. Algorithmic search for baseline minimum aberration designs. *Journal of Statistical Planning and Inference*, 149:172–182, 2014.

[20] Arden Miller and Boxin Tang. Using regular fractions of two-level designs to find baseline designs. *Statistica Sinica*, 26(2):745–759, 2016.

[21] Rahul Mukerjee and Boxin Tang. Optimal two-level regular designs under baseline parametrization via cosets and minimum moment aberration. *Statistica Sinica*, 26(3): 1001–1019, 2016.

[22] Ching-Shui Cheng. Some projection properties of orthogonal arrays. *The Annals of Statistics*, 23(4):1223–1233, 1995.

[23] Jonathan W Stallings and JP Morgan. General weighted optimality of designed experiments. *Biometrika*, 102(4):925–935, 2015.

[24] Ching-Shui Cheng. Orthogonal arrays with variable numbers of symbols. *Annals of Statistics*, 8.

[25] RL Rechtschaffner. Saturated fractions of 2^n and 3^n factorial designs. *Technometrics*, 9(4):569–575, 1967.

[26] Ching-Shui Cheng. *Theory of Factorial Design: Single-and Multi-Stratum Experiments.* CRC Press, 2014.

[27] Indra M. Chakravarti. Fractional replication in asymmetrical factorial designs and partially balanced arrays. *Sankhyā: The Indian Journal of Statistics*, 17(2):143–164, 1956.

[28] JN Srivastava and DV Chopra. On the characteristic roots of the information matrix of 2^m balanced factorial designs of resolution V with applications. *The Annals of Mathematical Statistics*, 42(2):722–734, 1971.

[29] JP Morgan, Subir Ghosh, and Angela M Dean. JN Srivastava and experimental design. *Journal of Statistical Planning and Inference*, 144:3–18, 2014.

[30] Ruwan Chamara Karunanayaka and Boxin Tang. Compromise designs under baseline parameterization. *Journal of Statistical Planning and Inference*, 190:32–38, 2017.

[31] George EP Box and Norman R Draper. A basis for the selection of a response surface design. *Journal of the American Statistical Association*, 54(287):622–654, 1959.

[32] Jerome Sacks and Donald Ylvisaker. Some model robust designs in regression. *Annals of Statistics*, 12(4):1324–1348, 1984.

[33] Emmanuel Vazquez and Julien Bect. Sequential search based on kriging: convergence analysis of some algorithms. *arXiv preprint arXiv:1111.3866*, 2011.

[34] Thomas J Santner, Brian J Williams, William I Notz, and Brain J Williams. *The Design and Analysis of Computer Experiments*, volume 1. Springer, 2003.

[35] Art B Owen. Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, 2(2):439–452, 1992.

[36] Boxin Tang. Orthogonal array-based latin hypercubes. *Journal of the American Statistical Association*, 88(424):1392–1397, 1993.

[37] Yuanzhen He and Boxin Tang. Strong orthogonal arrays and associated latin hypercubes for computer experiments. *Biometrika*, 100(1):254–260, 2013.

[38] Chang-Xing Ma, Kai-Tai Fang, and Dennis KJ Lin. A note on uniformity and orthogonality. *Journal of Statistical Planning and Inference*, 113(1):323–334, 2003.

[39] Petros Drineas, Michael W Mahoney, Shan Muthukrishnan, and Tamás Sarlós. Faster least squares approximation. *Numerische Mathematik*, 117(2):219–249, 2011.

[40] Petros Drineas, Michael W Mahoney, and Shan Muthukrishnan. Sampling algorithms for l_2 regression and applications. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, 1127–1136, 2006.

[41] Petros Drineas, Malik Magdon-Ismail, Michael W Mahoney, and David P Woodruff. Fast approximation of matrix coherence and statistical leverage. *The Journal of Machine Learning Research*, 13(1):3475–3506, 2012.

[42] Ping Ma and Xiaoxiao Sun. Leveraging for big data regression. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(1):70–76, 2015.

[43] Jack Kiefer. Optimum experimental designs. *Journal of the Royal Statistical Society: Series B (Methodological)*, 21(2):272–304, 1959.

[44] Chenlu Shi and Boxin Tang. Model-bobust subdata selection for big data. Submitted, 2021.

[45] Chenlu Shi and Boxin Tang. Construction results for strong orthogonal arrays of strength three. *Bernoulli*, 26(1):418–431, 2020.