Title
Evaluation and Construction of Space-Filling Designs Based on Stratification

Author
Tian, Ye

Publication Date
2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Evaluation and Construction of Space-Filling Designs Based on Stratification

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Ye Tian

2021

ABSTRACT OF THE DISSERTATION

Evaluation and Construction of Space-Filling Designs
Based on Stratification

by

Ye Tian
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2021
Professor Hongquan Xu, Chair

Space-filling designs are commonly used in computer experiments aiming to build statistical surrogate models. In this thesis, we propose a minimum aberration type criterion and the stratified L_2-discrepancy for evaluating space-filling properties of designs based on design stratification properties on various grids. The idea of stratification comes from the stratified orthogonality of strong orthogonal arrays. The space-filling criterion provides a systematic way of classifying and ranking space-filling designs including various types of strong orthogonal arrays and Latin hypercube designs according to the space-filling hierarchy principle. Strong orthogonal arrays of maximum strength are favorable under the proposed space-filling criterion. The stratified L_2-discrepancy assesses the projection properties of designs based on points stratification properties and can be tuned flexibly. Projection uniformity is considered with respect to all possible stratifications with proper weights. The stratified L_2-discrepancy includes the space-filling criterion as a special case, and is suitable for evaluating all kinds of designs with little curse of dimensionality. We further derive lower bounds for the stratified L_2-discrepancy and the space-filling pattern enumerator via defining a metric space that reveals the distance between points based on stratification. We show that generalized Hadamard matrices achieve the lower bounds and present a simple way to construct generalized Hadamard matrices via Galois fields. Comparisons between the optimal designs and other space-filling designs are illustrated.

The dissertation of Ye Tian is approved.

Weng Kee Wong

Yingnian Wu

Qing Zhou

Hongquan Xu, Committee Chair

University of California, Los Angeles

2021

To my mother,

I love you

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, Professor Hongquan Xu for assistance and dedicated involvement in every step my PhD study. He leads me to the stage of being a researcher from a student, gives me the courage to believe in myself and supports my work with great patience and expertise. He shows respect, consideration and tolerance always that I would not fear to make decisions. He is knowledgeable, insightful and enthusiastic to research and life. I could not have imagined having a better advisor and mentor than him. I will continue to try hard and stay diligent so that I could be an advisor like him someday.

I am very grateful to have Professors Weng Kee Wong, Yingnian Wu and Qing Zhou in my committee for their valuable guidance throughout my studies. Their classes are enjoyable and inspiring and make me think of the place I started when I was sitting in the Mathematical analysis classroom in my first year of collage. I hope I could give out lectures as good as they do. Additionally, I would like to express gratitude to Dr. Weng Kee Wong for his treasured support and brilliant ideas which are really influential in shaping my experiment methods and critiquing my results.

VITA

2012	B.S. in Statistics, Nankai University, Tianjin, China
2016–2018	Reader, Statistics Department, UCLA.
2019.6–9	Data analyst intern, Futurewei Technologies, Santa Clara, California
2020.6–9	Data Scientist intern, Amazon.com, Seattle, Washington.
2019–2021	Teaching Assistant/Associate, Statistics Department, UCLA.
2021	Research Assistant, Statistics Department, UCLA.

PUBLICATIONS

TIAN, Y. and XU, H. (2021). A Minimum Aberration Type Criterion for Selecting Space-Filling Designs. *Biometrika*. https://doi.org/10.1093/biomet/asab021.

CHAPTER 1

Introduction

Computer models are commonly used to simulate complex systems in science and engineering research. Computer experiments help to build up statistical surrogate models for computer models and make computation and optimization easier (Santner et al., 2003; Fang et al., 2006). Space-filling designs are widely chosen for computer experiments. As the name suggests, space-filling designs spread points uniformly in the design region. There are a number of different types of space-filling designs such as Latin hypercube designs and variations (Lin and Tang, 2015; Ba et al., 2015; Zhou and Xu, 2015; Xiao and Xu, 2017), maximin distance designs (Xiao and Xu, 2018; Wang et al., 2018; Li et al., 2020), and uniform designs (Fang et al., 2018). The curse of dimensionality makes it difficult for design points to cover the high-dimensional design region uniformly. In these cases, the space-filling properties of projection designs, phrased as projection properties, are evaluated. Projection properties in a low-dimensional space have been considered based on the assumption that the number of active factors is small. Joseph et al. (2015) and Sun et al. (2019) pointed out that maximin distance designs and uniform designs may have poor low dimensional projections which are not space-filling. Latin hypercubes based on orthogonal arrays (Tang, 1993; Xiao and Xu, 2018), maximum projection designs (Joseph et al., 2015) and uniform projection designs (Sun et al., 2019; Wang et al., 2020) are designs with good low-dimensional projection properties.

Orthogonality between columns naturally enhances projection properties of designs. Suppose we stratify the design region to subregions equally, designs are said to achieve stratification if they have an equal number of points in each subregion. Latin hypercube designs achieve stratifications in all univariate projections. Using orthogonal arrays of strength $t \geq 2$, Owen (1992) and Tang (1993) gave construction of space-filling designs that achieve stratifications in all t and lower dimensional projections. Motivated by digital nets in quasi-Monte Carlo methods, He and Tang (2013) proposed strong orthogonal arrays that are more space-filling than comparable ordinary orthogonal arrays. A strong orthogonal array of strength t achieves stratifications in all t-dimensional margins as comparable ordinary orthogonal arrays do. Further, it achieves stratifications on finer grids in all g-dimensional margins for any $g < t$. Latin hypercubes constructed from strong orthogonal arrays are more space-filling than those constructed from comparable ordinary orthogonal arrays in all g-dimensional projections for any $2 \leq g \leq t-1$. The stratified orthogonality of strong orthogonal arrays inspires a new way to assess design projection properties based on stratification.

There are plenty of follow-up work related to the strong orthogonal arrays. Characterization of strong orthogonal arrays of strength 3 is presented by He and Tang (2014) through the notion of semi-embeddability. Despite the great space-filling properties of strong orthogonal arrays of strength 3, the run size n is often too large. He et al. (2018) introduced strong orthogonal arrays of strength 2+ and presented construction results for such designs. Strong orthogonal arrays of strength $2+$ achieve the same 2-dimensional stratifications as strong orthogonal arrays of strength 3 while keeping run sizes small. Shi and Tang (2019) proposed methods to distinguish strong orthogonal arrays of strength $2+$ and 2 based on 3-dimensional and 2-dimensional projections, respectively. Zhou and Tang (2019) studied the construction of strong orthogonal arrays of strength $2+$ and $3-$ with column-orthogonality. Lin and Tang (2015) and Li et al. (2021) studied sliced strong orthogonal arrays and column-orthogonal nearly strong orthogonal arrays. Shi and Tang (2020) further proposed construction methods of strong orthogonal arrays of strength 3 with additional stratification properties. Despite all this progress, the important topic of design selection of strong orthogonal arrays has not been systematically addressed. As we will show, there are many strong orthogonal arrays of the same strength, yet they have quite different space-filling properties.

Orthogonality and stratification discussed above are for designs with a fixed number of levels. Uniform designs are the kind of space-filling designs that spread design points uniformly on unit hypercubes with low discrepancy (Fang et al., 2006, 2018). Discrepancy measures the deviation between the empirical distribution of the design and the theoretical uniform distribution. The famous Koksma–Hlawka inequality in quasi-Monte Carlo method shows that the overall mean model could benefit from designs with low star discrepancy

$$|\bar{y}(\mathcal{P}) - E(y)| \leq V(f)D^*(\mathcal{P})$$

where $\bar{y}(\mathcal{P})$ is the sample mean, $E(y)$ is the expectation, $V(f)$ is the total variation of the function f in the sense of Hardy and Krause (Hua and Wang, 1981) and $D^*(\mathcal{P})$ is the star discrepancy (Weyl, 1916) of design $\mathcal{P}$. The star discrepancy is a special case of star L_p-discrepancy when $p = \infty$ and it cannot be computed in polynomial time (Winker and Fang, 1997). The star L_2-discrepancy can be calculated with a simple formula in $O(n^2m)$ time where n is the number of points in design and m is the number of dimensions. However, projection property is not well measured for the star L_2-discrepancy. Proposed by Hickernell (1998), the family of generalized L_2-discrepancy adopts a general structure that considers the projection property in all subset dimensions. The generalized L_2-discrepancy can be defined as a norm in a reproducing kernel Hilbert space, where analytical formula of the discrepancies can be derived easily. The centered L_2-discrepancy and wrap-around L_2-discrepancy are the most widely used ones among the family. Both of the discrepancies share good geometric interpretations and simple computation formula. Nevertheless, as pointed out by Zhou et al. (2013) and He et al. (2020), the centered L_2-discrepancy suffers from curse of dimensionality and prefers points close to the center. The wrap-around L_2-discrepancy is not sensitive to location shift.

In Chapter 2, we will propose a minimum aberration type criterion for assessing the space-filling properties of designs based on design stratification properties on various grids. A space-filling hierarchy principle is proposed as a basic assumption of the criterion where stratifications on larger volume grids are preferred over stratifications on smaller volume grids. Stratification properties are characterized numerically by a vector called the space-filling pattern, each entry of which reveals how space-filling the design points are on stratifications with certain volume grids. We rank and select designs that sequentially minimize the entries of the space-filling pattern.

The space-filling criterion is applicable to designs with s^p levels where s, p are positive integers, including Latin hypercube designs, strong orthogonal arrays and variations. Connection can be seen between our criterion, the generalized minimum aberration criterion (Xu and Wu, 2001) and minimum G_2-aberration criterion (Tang and Deng, 1999). We show that the space-filling criterion is capable of selecting efficient space-filling designs for building statistical surrogate models via a simulation comparison study.

In Chapter 3, we will propose the stratified L_2-discrepancy to evaluate uniform designs based on stratification properties. The stratification scheme applied on the unit hypercube is flexibly determined by users. A stratified region is proposed to calculated the local projection discrepancies. The stratified L_2-discrepancy could be treated as a member of the generalized L_2-discrepancy family with specialty in the way projection properties are considered. Stratifications on various grids are assessed with allocated weights so that stratifications are considered in different importance levels. This enables the stratified L_2-discrepancy to adopt principles based in the demand of the users. Curse of dimensionality is eliminated in our discrepancy. We establish the connection between the space-filling pattern and the stratified L_2-discrepancy by the space-filling pattern enumerator. The space-filling criterion in Chapter 2 is a special case of the stratified L_2-discrepancy under the space-filling hierarchy principle. We provide examples and comparisons to show the advantages of the stratified L_2-discrepancy over other discrepancies.

In Chapter 4, we will derive lower bounds for the stratified L_2-discrepancy and the space-filling pattern enumerator. A distance metric, called the NRT-distance is defined to reveal the distance between points based on stratification. Both the stratified L_2-discrepancy and the space-filling pattern enumerator could be expressed based on NRT-distance. We find that a class of optimal designs that achieve the lower bounds are the generalized Hadamard matrices. There are fruitful results on the construction of generalized Hadamard matrices. We present a series of simple construction methods using the multiplication tables over Galois fields. Examples of optimal designs are given. We compare our optimal designs with other space-filling designs based on various discrepancies and sub-dimensional projection properties. The optimal designs show competitive performance under other discrepancies and outperform the rest of the space-filling designs in lower-dimensional projections.

CHAPTER 2

A Minimum Aberration Type Criterion for Selecting Space-Filling Designs

This chapter proposes a minimum aberration type criterion for classifying and selecting space-filling designs in a systematic way. The new criterion is inspired by the popular minimum aberration criterion and its extensions, which are widely used for selecting and ranking fractional factorial designs; see Mukerjee and Wu (2006), Wu and Hamada (2009) and Cheng (2014). The underlying assumption for the minimum aberration criterion is the effect hierarchy principle (Wu and Hamada, 2009): (i) lower-order effects are more likely to be important than higher-order effects; (ii) effects of the same order are equally likely to be important. However, the minimum aberration criterion and its extensions cannot be used to assess the space-filling properties of Latin hypercube designs and strong orthogonal arrays; for example, all Latin hypercubes of the same size, whether orthogonal array-based or not, have the same generalized wordlength pattern.

Instead of considering factorial effects, we take stratification properties into consideration when assessing space-filling properties. Our space-filling criterion is based on the following space-filling hierarchy principle: (i) stratifications on larger grids are more likely to be important than stratifications on smaller grids; (ii) stratifications on the same volume grids are equally likely to be important. We formalize the principle here although it was implicitly used in the development of strong orthogonal arrays. We carefully define the space-filling pattern to characterize the stratification properties on various grids according to this principle. The new space-filling criterion is to select the designs that sequentially minimize the space-filling pattern. The new criterion can be applied to a broader class of designs including aforementioned various strong orthogonal arrays and Latin hypercube designs. We further show that strong orthogonal arrays of maximum strength are favorable under the space-filling criterion and present examples to show that the new criterion can classify and rank designs of the same strength.

2.1 Notation and Backgrounds

Let $\mathbb{Z}_s = \{0, 1, \ldots, s-1\}$ be the ring of integers modulo s. An orthogonal array of strength t, denoted by OA$(n, m, s_1 \times \cdots \times s_m, t)$, is an $n \times m$ matrix whose entries of its jth column are taken from $\mathbb{Z}_{s_j}$ and in any t-column subarray, all possible level combinations appear equally often. If $s_1 = \cdots = s_m = s$, the orthogonal array is symmetric and can be written as OA(n, m, s, t). For a symmetric orthogonal array of strength t, the number of rows must be a multiple of s^t. Define $n = \lambda s^t$ where λ is called the index of the orthogonal array. Latin hypercubes are $n \times m$ matrices of which each column is strictly a permutation of n evenly spread levels, say $\{0, \ldots, n-1\}$. They are special orthogonal arrays of strength 1 with $\lambda = 1$.

A strong orthogonal array of n runs, m factors, s^t levels and strength t is an $n \times m$ matrix with entries from $\mathbb{Z}_{s^t}$ such that any subarray of g columns for any $1 \leq g \leq t$ can be collapsed into an OA$(n, g, s^{u_1} \times \cdots \times s^{u_g}, g)$ for any set of positive integers $\{u_1, \ldots, u_g\}$ satisfying $u_1 + \cdots + u_g = t$. Collapsing s^t levels into s^{u_j} levels is done by calculating $\lfloor a/s^{t-u_j} \rfloor$ for $a = 0, 1, \ldots, s^t - 1$, where $\lfloor x \rfloor$ denotes the largest integer not exceeding x. We denote this strong orthogonal array as SOA(n, m, s^t, t) with its properties called the stratified orthogonality. Similarly, the index λ of a strong orthogonal array is defined as $n = \lambda s^t$. If $\lambda = 1$, the corresponding strong orthogonal array is also a Latin hypercube. Strong orthogonal arrays are closely related to nets and sequences from quasi-Monte Carlo methods. Suppose an elementary interval in base s is an interval in $[0, 1)^m$ defined as $\prod_{j=1}^{m} \left[\frac{A_j}{s^{u_j}}, \frac{A_j+1}{s^{u_j}}\right)$ where nonnegative integer A_j, u_j satisfying $0 \leq A_j < s^{u_j}$. For a given dimension $s \geq 1$, an integer base $s \geq 2$, a positive integer k, and an integer w with $0 \leq w \leq k$, a point set of s^k points in $[0, 1)^m$ is called a (w, k, m)-net in base s if every elementary interval in base s of volume s^{w-k} contains exactly s^w points. Digital nets is a general framework for the construction of (w, k, m)-nets.

A design matrix with fixed levels can be seen as design points distributed in a grid space. An SOA(n, m, s^t, t) can be treated as n design points spreading within an s^{tm} hypercube. To assess the space-filling property of a design, especially in a high dimensional space, we could assess its low dimensional projections. As a result of stratified orthogonality, strong orthogonal arrays of strength t achieve stratification no matter in what way the design space is divided into s^t equal-volume grids from projection. For example, strong orthogonal arrays of strength 3 guarantee stratifications on s^3 grids in any one dimension, $s^2 \times s$ and $s \times s^2$ grids in any two dimensions, and $s \times s \times s$ grids in any three dimensions of the design region. Other examples include strong orthogonal arrays of strength $2+$ which achieve stratifications on s^2 grids in any one dimension, and $s^2 \times s$ and $s \times s^2$ grids in any two dimensions, strong orthogonal arrays of strength $3-$ which achieve all stratifications as of strength 3 except for s^3 grids in any one dimension as the total number of levels is only s^2. Stratified orthogonality guarantees good projection properties on finer grids.

To discuss a broader class of space-filling designs, we introduce the concept of general strong orthogonal arrays which include strong orthogonal arrays as special cases. A general strong orthogonal array of n runs, m factors, s^p levels and strength t, denoted by GSOA(n, m, s^p, t), is an $n \times m$ matrix with entries from $\mathbb{Z}_{s^p}$ such that any subarray of g columns for any $1 \leq g \leq t$ can be collapsed into an OA$(n, g, s^{u_1} \times \cdots \times s^{u_g}, g)$ for any set of positive integers $\{u_1, \ldots, u_g\}$ satisfying $u_1 + \cdots + u_g = t$ and $u_i \leq p$ for $i = 1, \ldots, g$. General strong orthogonal arrays of strength t achieve stratification no matter in what way the design space is divided into s^t equal-volume grids from projection. Strong orthogonal arrays have the constraint $t = p$, that is, a GSOA(n, m, s^t, t) is an SOA(n, m, s^t, t). Without this constraint, general strong orthogonal arrays include any design with s^p levels to the framework. Specifically, general strong orthogonal arrays of strength $t = 3$ and $p = 2$ are strong orthogonal arrays of strength $3-$, that is, a GSOA$(n, m, s^2, 3)$ is an SOA$(n, m, s^2, 3-)$. General strong orthogonal arrays with $p = 1$ are ordinary orthogonal arrays, that is, a GSOA(n, m, s^1, t) is an OA(n, m, s, t). There may be situations where s and p are not clear. In such situations we will explicitly state either s or p or both. In this chapter we focus on

Table 2.1: Mapping functions and weights for $x \in \mathbb{Z}_{2^3}$.

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$\rho(x)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	2
3	0	1	1	2
4	1	0	0	3
5	1	0	1	3
6	1	1	0	3
7	1	1	1	3

$p > 1$ even though the criterion and results apply for $p = 1$.

2.2 A Space-Filling Criterion

2.2.1 Characters

We first define a series of mapping functions f_i from $\mathbb{Z}_{s^p}$ to $\mathbb{Z}_s$. For $i = 1, \ldots, p$ and $x \in \mathbb{Z}_{s^p}$, let $f_i(x) = \lfloor x/s^{p-i} \rfloor \mod s$. Function $f_i(x)$ gives out the ith digit of x in the base-s numeral system. The set of mapping functions is a bijection function that expands x to a p-dimensional vector whose elements are from $\mathbb{Z}_s$. This expansion makes it possible to obtain information of x in each possible division. To transfer back, $x = \sum_{i=1}^{p} f_i(x) s^{p-i}$.

For $x \in \mathbb{Z}_{s^p}$, define weight $\rho(x) = p + 1 - \min\{i | f_i(x) \neq 0, i = 1, \ldots, p\}$ if $x \neq 0$ and $\rho(0) = 0$. The weight $\rho(x)$ is a generalization of the Hamming weight and represents the number of digits needed to express x in the base-s numeral system after wiping out all the leading zeros. As an example, Table 2.1 lists the mapping functions and the weights for all possible $x \in \mathbb{Z}_{2^3}$.

Table 2.2: Characters $\chi_u(x)$ for $u, x \in \mathbb{Z}_{2^3}$.

x	$\chi_0(x)$	$\chi_1(x)$	$\chi_2(x)$	$\chi_3(x)$	$\chi_4(x)$	$\chi_5(x)$	$\chi_6(x)$	$\chi_7(x)$
0	1	1	1	1	1	1	1	1
1	1	1	1	1	-1	-1	-1	-1
2	1	1	-1	-1	1	1	-1	-1
3	1	1	-1	-1	-1	-1	1	1
4	1	-1	1	-1	1	-1	1	-1
5	1	-1	1	-1	-1	1	-1	1
6	1	-1	-1	1	1	-1	-1	1
7	1	-1	-1	1	-1	1	1	-1

Define an inverse inner product between $u, x \in \mathbb{Z}_{s^p}$ as

$$\langle u, x \rangle = f_p(u) f_1(x) + \cdots + f_1(u) f_p(x) = \sum_{i=1}^{p} f_{p-i+1}(u) f_i(x).$$

As an illustration, suppose the entries of a design matrix are from $\mathbb{Z}_{2^3}$. Mapping functions for $u = 2$ and $x = 6$ are $(f_1(2), f_2(2), f_3(2)) = (0, 1, 0)$ and $(f_1(6), f_2(6), f_3(6)) = (1, 1, 0)$. The inverse inner product between u and x is $\langle 2, 6 \rangle = f_3(2) f_1(6) + f_2(2) f_2(6) + f_1(2) f_3(6) = 0 \cdot 1 + 1 \cdot 1 + 0 \cdot 0 = 1$.

For $u, x \in \mathbb{Z}_{s^p}$, define character $\chi_u(x) = \xi^{\langle u, x \rangle}$, where $\xi = e^{2\pi i/s}$ is the primitive sth root of unity. For simplicity, we use i for both $(-1)^{1/2}$ and an index. The meaning of i should be clear from the context. As an example, for $u = 2$ and $x = 6$, we have $\chi_2(6) = \xi^{\langle 2,6 \rangle} = (-1)^1 = -1$. Table 2.2 shows the values of all possible characters $\chi_u(x), u, x \in \mathbb{Z}_{2^3}$.

We can expand the definitions of weight and character to vectors over $\mathbb{Z}_{s^p}$. For $u = (u_1, \ldots, u_m) \in \mathbb{Z}_{s^p}^m$, the weight $\rho(u) = \sum_{i=1}^m \rho(u_i)$ is defined as the summation of individual weights, and the character $\chi_u(x) = \prod_{i=1}^m \chi_{u_i}(x_i)$ for any $x = (x_1, \ldots, x_m) \in \mathbb{Z}_{s^p}^m$ is defined as the tensor product of individual characters. For $u = (2, 3, 6), x = (6, 5, 4) \in \mathbb{Z}_{2^3}^3$, $\rho(u) = \sum_{i=1}^m \rho(u_i) = 2 + 2 + 3 = 7$, and $\chi_u(x) = \chi_2(6)\chi_3(5)\chi_6(4) = \xi^{\langle 2,6 \rangle + \langle 3,5 \rangle + \langle 6,4 \rangle} = (-1)^2 = 1$.

Let $\tau = s^{mp}$. Let $x_1, \ldots, x_\tau$ and let $u_1, \ldots, u_\tau$ denote all possible $x, u \in \mathbb{Z}_{s^p}^m$ in Yates order. Let $H = (\chi_{u_j}(x_i))$ be the $\tau \times \tau$ matrix of characters evaluated at all possible points in $\mathbb{Z}_{s^p}^m$. Both the first row and first column of H are vectors of ones. The matrix H is symmetrical because $\chi_u(x) = \chi_x(u)$. Table 2.2 shows the H matrix with $s = 2$, $p = 3$ and $m = 1$. When $s > 2$, H is a matrix of complex numbers. Let H^* be the conjugate transpose of H.

Theorem 2.1. *The character matrix H is symmetrical and orthogonal, that is, $H^{\mathrm{T}} = H$ and $H^*H = HH^* = \tau I$, where I is an identity matrix of order τ.*

2.2.2 Characteristics and Space-Filling Pattern

Let D be a design with n runs, m columns and s^p levels. Design D can be regarded as n points spreading in the design space $\mathbb{Z}_{s^p}^m$. There are in total τ distinct points in $\mathbb{Z}_{s^p}^m$. For each $x \in \mathbb{Z}_{s^p}^m$, let N_x be the number of times x appearing in D. If we ignore row orders, the design matrix can be represented uniquely by the frequency vector $N(D) = (N_{x_1}, \ldots, N_{x_\tau})$, where $x_1, \ldots, x_\tau$ are all of the distinct points in $\mathbb{Z}_{s^p}^m$ arranged in Yates order. We call this vector $N(D)$ the frequency representation of D.

For any $u \in \mathbb{Z}_{s^p}^m$, define $\chi_u(D) = \sum_{x \in D} \chi_u(x)$ where x is a row of D and the summation is over all rows of D. The set of characteristics of D is defined as $\chi(D) = (\chi_{u_1}(D), \ldots, \chi_{u_\tau}(D))$, where $u_1, \ldots, u_\tau$ are all distinct points in $\mathbb{Z}_{s^p}^m$ in Yates order. The set of all $\chi_u(D)$ fully characterizes the properties of D. The following theorem shows that the set of characteristics and the frequency representation are connected through H.

Theorem 2.2. *The set of characteristics and the frequency representation uniquely determine each other as follows: $\chi(D) = N(D)H$ and $N(D) = \tau^{-1}\chi(D)H^*$.*

Now we are ready to define the space-filling pattern. For $j = 0, \ldots, mp$, define

$$S_j(D) = n^{-2} \sum_{\rho(u)=j} |\chi_u(D)|^2 = n^{-2} \sum_{\rho(u)=j} \chi_u(D)\overline{\chi_u(D)}, \tag{2.1}$$

Table 2.3: A Latin hypercube (left) and an SOA$(8, 3, 8, 3)$ (right).

0	0	0	0	0	0
1	1	4	2	3	6
2	4	1	3	6	2
3	5	5	1	5	4
4	2	2	6	2	3
5	3	6	4	1	5
6	6	3	5	4	1
7	7	7	7	7	7

where the summation is over all $u \in \mathbb{Z}_{s^p}^m$ with $\rho(u) = j$ and $\overline{\chi_u(D)}$ is the complex conjugate of $\chi_u(D)$. It is easy to show $S_0(D) = 1$. The vector $(S_1(D), \ldots, S_{mp}(D))$ is called the space-filling pattern.

Example 2.1. *Table 2.3 lists an* 8×3 *Latin hypercube design* D *and an SOA*$(8, 3, 8, 3)$*. The Latin hypercube design is generated from an OA*$(8, 3, 2, 3)$ *according to Tang (1993) while the SOA*$(8, 3, 8, 3)$ *is from He and Tang (2014). Here is an illustration on how to calculate the space-filling pattern. The entries of the designs are from* $\mathbb{Z}_{2^3}$*. The mapping functions with weights and the table of characters are given in Tables 2.1 and 2.2. The collection of* $u \in \mathbb{Z}_{2^3}^3$ *with weight one is the set* $\{(1, 0, 0), (0, 1, 0), (0, 0, 1)\}$*, so*

$$S_1(D) = n^{-2} \sum_{\rho(u)=1} |\chi_u(D)|^2 = \frac{1}{64} \left(|\chi_{(1,0,0)}(D)|^2 + |\chi_{(0,1,0)}(D)|^2 + |\chi_{(0,0,1)}(D)|^2 \right) = 0.$$

The rest of the space-filling pattern can be calculated in a similar way. The sizes of sets $\{u : u \in \mathbb{Z}_{2^3}^3, \rho(u) = i\}$ *for* $i = 0, \ldots, 9$ *are* $1, 3, 9, 25, 42, 72, 104, 96, 96, 64$*, respectively. The sum of the sizes is the total number of possible points in* $\mathbb{Z}_{2^3}^3$*. The space-filling pattern of the Latin hypercube design is* $(0, 0, 3, 5, 9, 16, 10, 12, 8)$*. Note that* $S_i(D) = 0, i = 1, 2$ *as all* $\chi_u(D) = 0$ *for* u *with* $0 < \rho(u) \leq 2$*.* $S_3(D) = 3$ *implies that not all* $\chi_u(D)$ *are zeros for* u *of weight 3. Specifically,* $\chi_u(D) = 8$ *for* $u = (1, 2, 0)$*,* $(1, 0, 2)$ *or* $(2, 1, 0)$*. Furthermore, the space-filling pattern for the*

SOA$(8,3,8,3)$ *is* $(0,0,0,12,6,13,12,12,8)$. *For both designs,* $\sum_{j=1}^{9} S_j(D) = 63$.

The following theorem shows that the space-filling pattern captures the strength of a general strong orthogonal array.

Theorem 2.3. *A general strong orthogonal array* D *has strength* t *if and only if* $S_j(D) = 0$ *for* $1 \leq j \leq t$.

Theorem 2.3 establishes the connection between stratified orthogonality and the space-filling pattern. If the first j elements of the space-filling pattern are zeros, the general strong orthogonal array achieves stratification on any s^j grids from projection. For example, $S_1(D) = 0$ guarantees that there is an equal number of design points when the design region is cut into s equal-volume grids in any one dimension. If $S_1(D) = S_2(D) = 0$, there is an equal number of design points in any s^2 equal-volume grids cut by projection: either s^2 grids of any one dimension or $s \times s$ grids of any two dimensions.

2.2.3 A minimum aberration type space-filling criterion

The space-filling pattern describes space-filling properties sequentially when the design is projected into coarse margins to finer margins. The properties are evaluated in the cluster of projections with respect to the volume of grids instead of dimensions. The space-filling hierarchy principle suggests that stratifications on larger grids are more likely to be important than stratifications on smaller grids. The minimum aberration type space-filling criterion is to select designs that sequentially minimize the space-filling pattern $S_j(D)$ for $j = 1, \ldots, mp$. Here is a formal definition.

Definition 2.1. *Suppose that designs* D_1 *and* D_2 *have space-filling patterns* $(S_1(D_1), \ldots, S_{mp}(D_1))$ *and* $(S_1(D_2), \ldots, S_{mp}(D_2))$*, respectively. If* $S_j(D_1) = S_j(D_2)$ *for* $j = 1, \ldots, l$*, and* $S_{l+1}(D_1) < S_{l+1}(D_2)$*, then* D_1 *is more space-filling than* D_2*. Design* D_1 *is the most space-filling if there is no other design that is more space-filling than* D_1*.*

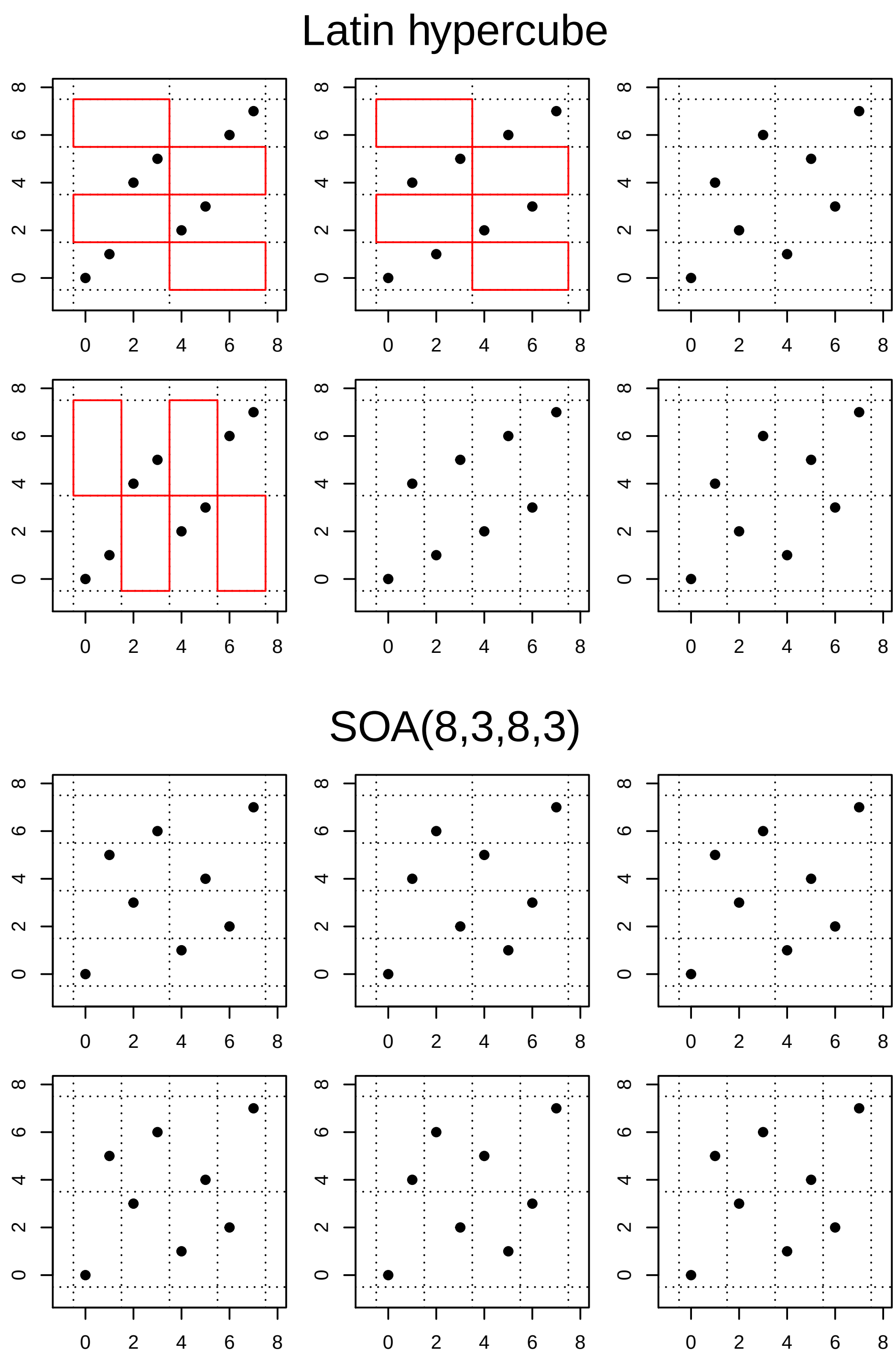

Figure 2.1: Projection plots: 2×4 (upper) and 4×2 (lower).

Example 2.2. *Recall that the space-filling patterns of the Latin hypercube and SOA*$(8,3,8,3)$ *in Table 2.3 are* $(0,0,3,4,9,16,10,12,8)$ *and* $(0,0,0,12,6,13,12,12,8)$*, respectively. According to the space-filling criterion, the SOA*$(8,3,8,3)$ *is more space-filling as its* $S_3 = 0$ *compared with* $S_3 = 3$ *for the Latin hypercube. Both designs achieve stratifications in each dimension,* 2×2 *grids in 2 dimensions and* $2\times 2\times 2$ *grids in 3 dimensions. Figure 2.1 presents their* 2×4 *and* 4×2 *projection plots. The projection dimensions are 1& 2, 1 & 3 and 2 & 3 from left to right. The strong orthogonal array achieves stratifications as each grid has exactly one design point. On the other hand, the Latin hypercube does not have an equal number of points in all* 2×4 *and* 4×2 *grids. We highlight the grids that do not have any design points. Our ranking agrees with the conclusion in He and Tang (2013) that Latin hypercubes based on strong orthogonal arrays are more space-filling than comparable Latin hypercubes based on orthogonal arrays.*

Values of the space-filling pattern quantify the stratified orthogonality. Leading zeros indicate stratifications on certain number of equal-volume grids. The first non-zero element reveals how space-filling the design performs when projected to the specific number of grids. Specifically, the $S_j(D)$ value reveals how uniform the points are distributed when the design is projected to s^j grids. When two general strong orthogonal arrays have the same strength, the space-filling criterion selects the design that is more space-filling in the next finer projections.

Theorem 2.4. *For a design* D *with* n *runs,* m *columns and* s^p *levels. The sum of its space-filling pattern has a lower bound:*

$$\sum_{j=1}^{mp} S_j(D) \geq \frac{s^{mp}}{n} - 1. \tag{2.2}$$

The equality holds if and only if D *has no replicated points.*

Theorem 2.4 shows that the space-filling pattern characterizes whether a design has replicated points. Designs without replicated points are more space-filling and have smaller $\sum_{j=1}^{mp} S_j(D)$ than designs with replicated points. Latin hypercube designs do not have replicated points, so the equality in (2.2) always holds for Latin hypercube designs.

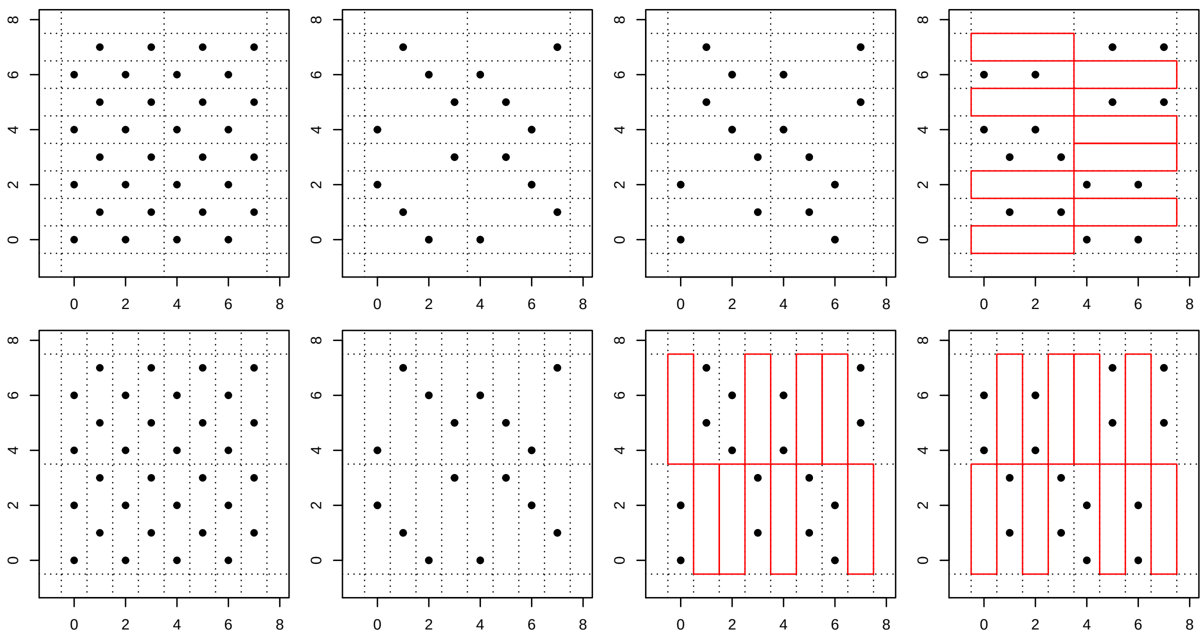

Figure 2.2: Projection plots of designs D_1, D_2, D_3, D_4 (from left to right): 2×8 (upper) and 8×2 (lower).

Example 2.3. *Consider the SOA$(32, 9, 8, 3)$ listed in Shi and Tang (2020). We search over all 36 two-column subarrays and find a total of four distinct space-filling patterns, which differ in (S_4, S_5, S_6). The four distinct patterns of (S_4, S_5, S_6) are $(0, 0, 1)$ for columns (1, 2) denoted as D_1, $(0, 2, 1)$ for columns (2, 6) as D_2, $(1, 1, 1)$ for columns (1, 8) as D_3, and $(2, 0, 1)$ for columns (1, 7) as D_4, respectively. For D_1, there are no replicated points and the equality in* (2.2) *holds as $\sum_{j=1}^{6} S_j(D_1) = 1$. For other three designs, there are only 16 distinct points and $\sum_{j=1}^{6} S_j(D) = 3$. The space-filling criterion ranks D_1 the best, followed by D_2, D_3 and D_4. Figure 2.2 shows the 2×8 and 8×2 projection plots of these four designs. For D_1 and D_2, $S_4 = 0$ guarantees stratification on any 2^4 grids. Design D_1 with $S_5 = 0$ achieves stratifications on 4×8 and 8×4 grids whereas D_2 does not. Between D_3 and D_4, D_3 is more space-filling than D_4 and achieves stratification on 2×8 grids, whereas D_4 does not achieve stratifications on 2×8 and 8×2 grids.*

Example 2.4. *Table 1 of Sun et al. (2019) lists four 25×3 Latin hypercubes: a uniform de-*

sign, a maximin distance design, a maximum projection design and a uniform projection design. We can compare and rank them using the space-filling criterion. Their space-filling patterns are $(S_1, S_2, S_3, \ldots)$ = $(0, 0.64, 26.08, \ldots)$, $(0, 1.84, 22.48, \ldots)$, $(0, 0.96, 25.12, \ldots)$, $(0, 0, 28, \ldots)$, *respectively. The uniform projection design has strength 2 whereas other three designs have strength 1. This agrees with the scatter plots in Fig. 1 of Sun et al. (2019). Among these four designs, the uniform projection design is the most space-filling, and the maximin distance design is the least space-filling. The uniform design is more space-filling than the maximum projection design. The ranking of these four designs under the space-filling criterion is consistent with the ranking under the uniform projection criterion used by Sun et al. (2019).*

2.2.4 Connections with other criteria

Here we explore the connection between the space-filling criterion and other criteria. When $s = 2$ and $p = 1$, the characteristics $\chi_u(D)$ are called J-characteristics by Tang and Deng (1999) and Tang (2001). For regular designs, any two factorial effects either can be estimated independently of each other or are fully aliased. Designs that do not possess this property are called nonregular designs. For two-level nonregular designs, the space-filling pattern defined in (2.1) coincides with the generalized wordlength pattern and the space-filling criterion is equivalent to the minimum G_2-aberration criterion proposed by Tang and Deng (1999).

For general $s \geq 2$ and $p = 1$, the set of characters $\{\chi_u; u \in \mathbb{Z}_s\}$ forms the normalized orthogonal contrasts (Xu and Wu, 2001). For $u = (u_1, \ldots, u_m) \in \mathbb{Z}_s^m$, the weight $\rho(u)$ is the Hamming weight of u, i.e., the number of nonzero elements of u. For a design D with n runs, m factors and s levels, Xu and Wu (2001) defined the generalized wordlength pattern $(A_1(D), \ldots, A_m(D))$ where

$$A_j(D) = n^{-2} \sum_{\rho(u)=j} |\chi_u(D)|^2. \tag{2.3}$$

The generalized wordlength pattern reveals the aliasing structure of the design. The $A_j(D)$ measures the overall aliasing between all j-factor interactions and the intercept. It also measures the overall aliasing between all $(j-1)$-factor interactions and all main effects. The generalized minimum aberration criterion proposed by Xu and Wu (2001) is to sequentially minimize the elements in the generalized wordlength pattern.

The definition of $A_j(D)$ in (2.3) is a special case of the definition of $S_j(D)$ in (2.1) with $p = 1$, so the generalized wordlength pattern is also a special case of the space-filling pattern. As a result, the space-filling criterion is more general than the generalized minimum aberration criterion.

Despite the similarity of the definitions, it is important to notice the differences between these two concepts. The generalized wordlength pattern and generalized minimum aberration criterion are defined for selecting factorial designs under an ANOVA model. The generalized wordlength pattern considers the aliasing among factorial effects whereas the space-filling pattern considers the stratification properties of the designs. The generalized minimum aberration criterion treats the s levels as nominal symbols so that permuting levels for any column does not alternate the generalized wordlength pattern. In contrast, the space-filling criterion treats the s^p levels as numerical values so that permuting levels for any column may alternate design stratification structure and the space-filling pattern. For example, consider the two designs in Table 2.3. When we view them as ordinary 8-level factorial designs with $s = 8$ and $p = 1$, both designs have the same generalized wordlength pattern $(A_1, A_2, A_3) = (0, 21, 42)$. In contrast, when we view them as general strong orthogonal arrays with $s = 2$ and $p = 3$, they have different space-filling patterns and different strengths; see Examples 2.1, 2.2 and Fig. 2.1.

Shi and Tang (2019) considered design selection for strong orthogonal arrays with strength two plus. The criterion they used is equivalent to the minimization of $S_3(D)$. Their criterion works only for strong orthogonal arrays constructed from regular designs whereas our criterion is more general and works for any type of general strong orthogonal arrays.

2.3 Applications and Simulation Comparisons

We first apply the space-filling criterion for selecting and ranking designs. Shi and Tang (2020) considered constructions of strong orthogonal arrays of strength 3. They listed an SOA$(32, m, 8, 3)$

Table 2.4: Number of distinct space-filling patterns.

	Number of columns							
Design	2	3	4	5	6	7	8	9
SOA$(32, 7, 8, 3)$	2	4	4	2	1	1	-	-
SOA$(32, 8, 8, 3)$	3	5	8	5	3	1	1	-
SOA$(32, 9, 8, 3)$	4	23	92	121	83	35	7	1

for $m = 7, 8, 9$. We conduct an exhaustive search over all subarrays of these three designs. Table 2.4 presents the numbers of distinct space-filling patterns for all m-column subarrays of these three designs for $m = 2, ..., 9$. The numbers of distinct space-filling patterns vary substantially for different designs. For either SOA$(32, 7, 8, 3)$ or SOA$(32, 8, 8, 3)$, there are only a small number of distinct space-filling patterns. Subarrays of these designs are clustered with similar space-filling properties. However, for SOA$(32, 9, 8, 3)$, there are a large number of distinct space-filling patterns. For example, there are 121 distinct space-filling patterns among the total 126 5-column subarrays of SOA$(32, 9, 8, 3)$. For $m = 6, 7, 8$, the numbers of distinct space-filling pattern are 83, 35, 7, which are just 1 or 2 less than the total numbers of all subarrays, 84, 36, 9, respectively. Almost all subarrays have different space-filling properties.

Tables 2.5-2.7 list the best space-filling patterns and sets of representative column indices from these three designs. For $m = 2$, the best designs have strength 5, so they achieve stratifications on 8×4 and 4×8 grids. For $m = 3, 4$, the best designs have strength 4, so they are strong orthogonal arrays of strength $4-$, analogous to the definition of strong orthogonal arrays of strength $3-$, and achieve stratifications on 4×4, 8×2, 2×8, $4 \times 2 \times 2$, $2 \times 4 \times 2$, and $2 \times 2 \times 4$ grids, and as well as $2 \times 2 \times 2 \times 2$ grids for $m = 4$. For $m =$ 5–9, the best designs have strength 3. When $m =$ 4–8, the most space-filling design is either from SOA$(32, 7, 8, 3)$ or SOA$(32, 8, 8, 3)$. This indicates that neither SOA$(32, 7, 8, 2)$ nor SOA$(32, 8, 8, 3)$ is a subarray of SOA$(32, 9, 8, 3)$. In the supplementary, we list all possible space-filling patterns of m-column subarrays from SOA$(32, 7, 8, 3)$, SOA$(32, 8, 8, 3)$ and SOA$(32, 9, 8, 3)$ for $m = 2, ..., 9$.

Table 2.5: Best space-filling designs from SOA$(32, 7, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1, -	1, 4	4
3	0, 1, 7, 3	1, 4, 7	4
4	0, 9, 19, 11	1, 2, 4, 7	4
5	1, 22, 40, 40	1, 2, 3, 4, 6	3
6	3, 42, 83, 104	1, 2, 3, 4, 5, 6	3
7	7, 70, 161, 224	1, 2, 3, 4, 5, 6, 7	3

Table 2.6: Best space-filling designs from SOA$(32, 8, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1, -	1, 4	5
3	0, 1, 7, 3	1, 4, 6	4
4	0, 8, 20, 12	1, 2, 3, 5	4
5	3, 17, 44, 38	1, 2, 3, 5, 8	3
6	7, 36, 79, 108	1, 2, 3, 4, 5, 8	3
7	13, 62, 143, 248	1, 2, 3, 4, 5, 6, 7	3
8	22, 96, 252, 496	1, 2, 3, 4, 5, 6, 7, 8	3

Table 2.7: Best space-filling designs from SOA$(32, 9, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1, -	1, 2	5
3	0, 1, 7, 3	1, 5, 9	4
4	0, 9, 19, 11	3, 7, 8, 9	4
5	3, 16, 42, 46	1, 5, 6, 8, 9	3
6	8, 26, 89, 121	1, 4, 5, 6, 8, 9	3
7	15, 52, 145, 278	1, 2, 4, 5, 6, 8, 9	3
8	27, 80, 248, 546	1, 2, 3, 4, 5, 6, 8, 9	3
9	42, 124, 400, 976	1, 2, 3, 4, 5, 6, 7, 8, 9	3

We next evaluate the performance of general strong orthogonal arrays and compare them with other types of space-filling designs in building statistical surrogate models. We conduct simulations and generate data from the 8-dimensional borehole function, which has been used by Fang et al. (2006), Chen et al. (2016) and many others. We apply a log-transformation on the response as suggested by Fang et al. (2006). We fit a Gaussian process model with a constant mean and the Gaussian correlation function to approximate the borehole function. To measure the prediction error, we use the normalized root mean square error, that is,

$$\text{Normalized RMSE} = \left[\frac{N^{-1}\sum_{i=1}^{N}\{\hat{y}(x_i) - y(x_i)\}^2}{N^{-1}\sum_{i=1}^{N}\{\bar{y} - y(x_i)\}^2}\right]^{1/2}$$

where $\{x_1, \ldots, x_N\}$ is a set of N test data points, $y(x_i)$ is the true response at x_i, $\hat{y}(x_i)$ is the predicted response from the Gaussian process model, and $\bar{y}$ is the mean response of the data used to build the model. We generate a test dataset using a random Latin hypercube design with $N = 10,000$ runs.

We consider two SOA$(32, 8, 8, 3)$ according to Tables 2.6 and 2.7, which have $S_4 = 22$ and 27, respectively. We also generate random Latin hypercube designs from these two 8-level designs by expanding 8 levels to 32 levels following Tang (1993). These Latin hypercube designs are GSOA$(32, 8, 32, 3)$ and have the same S_4 values as the corresponding original SOA$(32, 8, 8, 3)$ while their S_5 values may vary. We also consider other four types of space-filling designs: maximin Latin hypercube designs, maximum projection Latin hypercube designs, uniform designs, and densest packing-based maximum projection designs. The maximin Latin hypercube design and the maximum projection Latin hypercube design are generated using R packages SLHD (Ba et al., 2015) and MaxPro (Joseph et al., 2015), respectively. We generate an 8-level and a 32-level uniform design using R package UniDOE (Zhang et al., 2018). The densest packing-based maximum projection design is generated using R package LatticeDesign (He, 2020). All designs have 32 runs and 8 columns, and each variable is scaled to $[0, 1]$. Given any design, we consider permuting column labels and reflecting within columns for a random subset of inputs, which do not change the design's geometrical structure and space-filling pattern. Figure 2.3 shows the normalized root mean square errors from 1,000 random permutations and reflections for each design. The SOA$(32, 8, 8, 3)$ with $S_4 = 22$ and its associated Latin hypercube designs clearly outperform other designs, including the SOA$(32, 8, 8, 3)$ with $S_4 = 27$ and its associated Latin hypercube designs. Our new space-filling criterion is capable of selecting efficient space-filling designs for building statistical surrogate models. The simulation also shows that designs with good asymptotic properties such as the densest packing-based designs and maximin designs may not work well when the run sizes are not large.

2.4 Concluding Remarks

This chapter introduces a minimum aberration type space-filling criterion for classifying and selecting space-filling designs. The new criterion selects designs based on the proposed space-filling hierarchy principle, that is, it prefers designs that achieve stratifications on larger grids to smaller grids. The space-filling pattern is defined to characterize the stratification properties on various grids. Our criterion works for any design with s^p levels. It favors strong orthogonal arrays of maximum strength and is also capable to distinguish strong orthogonal arrays with the same strength.

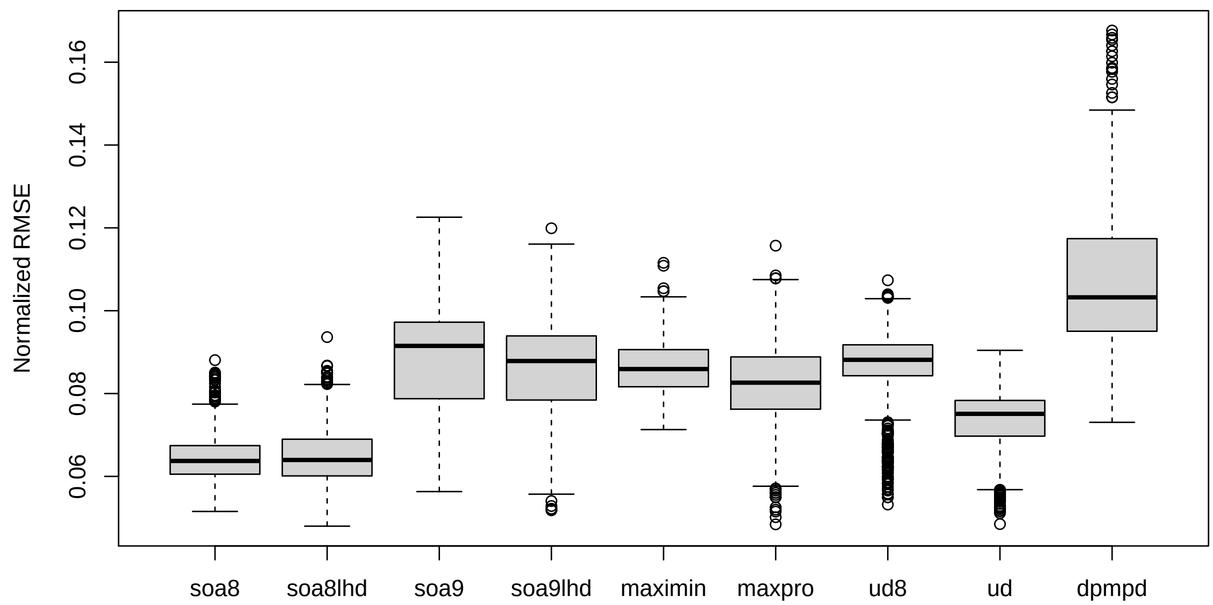

Figure 2.3: Normalized root mean square errors for SOA$(32, 8, 8, 3)$ with $S_4 = 22$ (soa8) and its associated Latin hypercube designs (soa8lhd), SOA$(32, 8, 8, 3)$ with $S_4 = 27$ (soa9) and its associated Latin hypercube designs (soa9lhd), maximin Latin hypercube designs (maximin), maximum projection Latin hypercube designs (maxpro), 8-level uniform designs (ud8), 32-level uniform designs (ud), and densest packing-based maximum projection designs (dpmpd).

We further demonstrate that general strong orthogonal arrays selected by our criterion have competitive performance among various types of space-filling designs in building statistical surrogate models.

This chapter opens many new research areas in this topic. The new space-filling criterion has a clear geometrical meaning. And it is helpful to have additional statistical justifications on the performance and properties of the criterion. The generalized minimum aberration criterion is closely related to various uniformity measures and the maximin criterion when all possible level permutations are considered (Tang et al., 2012; Zhou and Xu, 2014; Fang et al., 2018). It is interesting to investigate whether these connections can be extended to the new space-filling criterion. One difficulty is that we have to restrict level permutations in order to keep the space-filling pattern invariant. The calculation of the space-filling pattern by the definition (2.1) is tedious for large designs. A future research is to find an efficient calculation method to support the use of our criterion. For those designs with the same space-filling pattern, it is worth considering the definition of isomorphism in terms of space-filling properties. Developing a systematic method for constructing optimal general strong orthogonal arrays could be a big topic. We hope to develop more theoretical results about general strong orthogonal arrays in the future.

2.5 Appendix: Proofs

We provide properties of the characteristics and proofs of the theorems in this section.

Proof of Theorem 2.1. First consider $p = 1$. For any $u, v \in \mathbb{Z}_s$,

$$\sum_{x\in\mathbb{Z}_s} \chi_u(x)\overline{\chi_v(x)} = \sum_{x\in\mathbb{Z}_s} \xi^{ux}\xi^{-vx} = \sum_{x\in\mathbb{Z}_s} \xi^{(u-v)x} = s\delta_{u,v},$$

where $\xi = e^{2\pi i/s}$, $\overline{\chi_v(x)}$ is the complex conjugate of $\chi_v(x)$, and $\delta_{u,v}$ is 1 when $u = v$ and 0 otherwise.

Next consider $p \geq 1$. For any $u, x \in \mathbb{Z}_{s^p}$, $\chi_u(x) = \prod_{i=1}^{p} \xi^{f_{p-i+1}(u)f_i(x)}$. Then for any $u, v \in$

$\mathbb{Z}_{s^p}$,

$$\sum_{x\in\mathbb{Z}_{s^p}} \chi_u(x)\overline{\chi_v(x)} = \sum_{x\in\mathbb{Z}_{s^p}} \prod_{i=1}^{p} \xi^{[f_{p-i+1}(u)-f_{p-i+1}(v)]f_i(x)}.$$

As x varies over $\mathbb{Z}_{s^p}$, $f_i(x)$ varies over $\mathbb{Z}_s$ for $i = 1, \dots, p$. So we have

$$\sum_{x\in\mathbb{Z}_{s^p}} \chi_u(x)\overline{\chi_v(x)} = \prod_{i=1}^{p} \sum_{f_i(x)\in\mathbb{Z}_s} \xi^{[f_{p-i+1}(u)-f_{p-i+1}(v)]f_i(x)} = \prod_{i=1}^{p} s\delta_{f_{p-i+1}(u),f_{p-i+1}(v)} = s^p\delta_{u,v}.$$

Now for $x = (x_1, \dots, x_m)$, $u = (u_1, \dots, u_m)$ and $v = (v_1, \dots, v_m)$ from $\mathbb{Z}_{s^p}^m$,

$$\begin{aligned}\sum_{x\in\mathbb{Z}_{s^p}^m} \chi_u(x)\overline{\chi_v(x)} &= \sum_{x\in\mathbb{Z}_{s^p}^m} \prod_{j=1}^{m} \chi_{u_j}(x_j)\overline{\chi_{v_j}(x_j)} = \prod_{j=1}^{m} \sum_{x_j\in\mathbb{Z}_{s^p}} \chi_{u_j}(x_j)\overline{\chi_{v_j}(x_j)} \\ &= \prod_{j=1}^{m} s^p\delta_{u_j,v_j} = s^{mp}\delta_{u,v} = \tau\delta_{u,v}.\end{aligned}$$

This proves $HH^* = \tau I$, which also implies $H^*H = \tau I$. □

Proof of Theorem 2.2. Simple algebra connects the set of characteristics with the frequency representation by

$$\chi_u(D) = \sum_{x\in D} \chi_u(x) = \sum_{x\in\mathbb{Z}_{s^p}^m} N_x\chi_u(x).$$

Thus, the inner product between $N(D)$ and columns of H produces the set of characteristics, that is, $\chi(D) = N(D)H$. By Theorem 2.1, we have $N(D) = \tau^{-1}\chi(D)H^*$. □

To prove Theorem 2.3, we need to consider properties of the characteristics and its relation to projection designs. To facilitate the discussion of projection designs, we introduce some additional notation. Let $W = \{0, \dots, p\}^m$ be the weight set containing all possible weight patterns for $u \in \mathbb{Z}_{s^p}^m$. For any $w = (w_1, \dots, w_m) \in W$, let $\|w\|_1 = \sum_{i=1}^m w_i$ and $U_w = \{u : u \in \mathbb{Z}_{s^p}^m, \rho(u_1) \le w_1, \dots, \rho(u_m) \le w_m\}$. It is obvious that $\rho(u) \le \|w\|_1$ if $u \in U_w$. Let $D = (d_1, \dots, d_m)$ be a GSOA(n, m, s^p, t), where d_i is the ith column of D. Let $D_w = (\lfloor d_1/s^{p-w_1}\rfloor, \dots, \lfloor d_m/s^{p-w_m}\rfloor)$ be the collapsed projection design of D onto $s^{w_1} \times \cdots \times s^{w_m}$ margins. If D_w is a full design with an equal number of replicates, design D achieves stratification on $s^{w_1} \times \cdots \times s^{w_m}$ grids.

Examples of collapsed projection designs are given in the following. Suppose design $D = (d_1, d_2, d_3)$ is the SOA$(8, 3, 8, 3)$ in Table 2.3 with $s = 2$ and $p = 3$. For $w = (1, 2, 3)$, the three columns of $D_w = (\lfloor d_1/4 \rfloor, \lfloor d_2/2 \rfloor, \lfloor d_3/1 \rfloor)$ have 2, 4, and 8 levels, respectively. For $w = (0, 3, 3)$, D_w is the projection design of D to the second and the third dimensions, each with 8 levels. The first dimension of D_w is negligible because all entries are zeros. In the following we call collapsed projection designs as projection designs.

Lemma 2.1. *For any $w \in W$, the projection design D_w is uniquely characterized by $\chi_u(D)$ for all $u \in U_w$.*

Proof. Let $D = (d_1, \ldots, d_m)$. When $u \in U_w$, $\rho(u_j) \le w_j$ for $j = 1, \ldots, m$. If $u_j \ne 0$, $\rho(u_j) = p+1-\min\{i | f_i(u_j) \ne 0\}$, so $\min\{i | f_i(u_j) \ne 0\} \ge p+1-w_j$ and $f_i(u_j) = 0$ for $i = 1, \ldots, p-w_j$. If $u_j = 0$, $f_i(u_j) = f_i(0) = 0$ for $i = 1, \ldots, p$. Then

$$\chi_u(D) = \sum_{x \in D} \chi_u(x) = \sum_{x \in D} \prod_{j=1}^{m} \xi^{\sum_{i=1}^{p} f_{p-i+1}(u_j) f_i(x_j)} = \sum_{x \in D} \prod_{j=1}^{m} \xi^{\sum_{i=1}^{w_j} f_{p-i+1}(u_j) f_i(x_j)}.$$

This implies that for each column d_j, among the p mapped columns $(f_1(d_j), \ldots, f_p(d_j))$, only the first w_j columns are involved in the calculation of $\chi_u(D)$. Furthermore, this set of w_j mapped columns uniquely determines the collapsed column $\lfloor d_j/s^{p-w_j} \rfloor$. As a result, we have $D_w = (\lfloor d_1/s^{p-w_1} \rfloor, \ldots, \lfloor d_m/s^{p-w_m} \rfloor)$ uniquely determined by $\chi_u(D), u \in U_w$. □

Lemma 2.2. *For any $w \in W$, the projection design D_w is a full design with an equal number of replicates if and only if $\chi_u(D) = 0$ for all $u \in U_w$ with $\rho(u) > 0$.*

Proof. For clarity, we only prove the result for the special case when $D_w = D$. The proof for general cases is similar but involves more complicated notation.

If D is a full design with an equal number of replicates, $N(D) = \lambda e$ where λ is the number of replicates and e is a row vector of ones. By Theorem 2.2, $\chi(D) = N(D)H = \lambda e H$. Because both the first row and the first column of H (and H^*) are a vector of ones, we have $\chi(D) = (\lambda\tau, 0, \ldots, 0)$.

On the other hand, if $\chi(D) = (\lambda, 0, \ldots, 0)$ for an integer λ, by Theorem 2.2, $N(D) = \tau^{-1}\chi(D)H^* = \tau^{-1}(\lambda, \ldots, \lambda) = \lambda\tau^{-1}e$. Then D is a full design with an equal number of replicates. $\square$

General strong orthogonal arrays of strength t achieve stratification when projected to any s^t grids. Thus, any projection design to s^t grids is a full design with an equal number of replicates. Applying Lemma 2.2, we have

Lemma 2.3. *Design D is a GSOA(n, m, s^p, t) if and only if $\chi_u(D) = 0$ for all $u \in \mathbb{Z}_{s^p}^m$ with $0 < \rho(u) \leq t$.*

Proof. Suppose that D is a GSOA(n, m, s^p, t). For any $w \in W$ with $\|w\|_1 = t$, D_w is a projection design of D to $s^{w_1} \times \cdots \times s^{w_m} = s^t$ grids. Since D has strength t, D_w is a full design with an equal number of replicates. By Lemma 2.2, $\chi_u(D) = 0$ for all $u \in U_w$ with $\rho(u) > 0$. Because $\bigcup_{w \in W, \|w\|_1 = t} U_w = \{u \in \mathbb{Z}_{s^p}^m : \rho(u) \leq t\}$, $\chi_u(D) = 0$ for all $u \in \mathbb{Z}_{s^p}^m$ with $0 < \rho(u) \leq t$.

On the other hand, suppose that $\chi_u(D) = 0$ for $u \in \mathbb{Z}_{s^p}^m$ with $0 < \rho(u) \leq t$. Then for any $w \in W$ with $\|w\|_1 = t$, $\chi_u(D) = 0$ for all $u \in U_w$ with $\rho(u) > 0$. By Lemma 2.2, D_w is a full design with an equal number of replicates. This is true for all projection designs D_w with $\|w\|_1 = t$; therefore, D is a GSOA(n, m, s^p, t). $\square$

Finally, Theorem 2.3 is a natural result of Lemma 2.3 and the definition (2.1).

Proof of Theorem 2.4. By the definition of $S_j(D)$ in (2.1), we have

$$\sum_{j=0}^{mp} S_j(D) = n^{-2} \sum_{u \in \mathbb{Z}_{s^p}^m} |\chi_u(D)|^2 = n^{-2}\chi(D)\overline{\chi(D)}^{\mathrm{T}}.$$

By Theorem 2.2, $\chi(D)\overline{\chi(D)}^{\mathrm{T}} = N(D)HH^*N(D)^{\mathrm{T}} = \tau N(D)N(D)^{\mathrm{T}}$. For any $x \in \mathbb{Z}_{s^p}^m$, N_x is a nonnegative integer, so $N_x(N_x - 1) \geq 0$ with equality if and only if $N_x = 0$ or 1. Thus, $N(D)N(D)^{\mathrm{T}} = \sum_{x \in \mathbb{Z}_{s^p}^m} N_x^2 \geq \sum_{x \in \mathbb{Z}_{s^p}^m} N_x = n$, where the equality holds if and only if every N_x is either 0 or 1 for every $x \in \mathbb{Z}_{s^p}^m$. Therefore, $\sum_{j=0}^{mp} S_j(D) \geq \tau/n$ with equality if and only if D has no replicated points. Finally, the result follows from the fact $S_0(D) = 1$. $\square$

2.6 Appendix: Tables

The search results of all possible space-filling patterns of m-column subarrays from SOA$(32, 7, 8, 3)$, SOA$(32, 8, 8, 3)$ and SOA$(32, 9, 8, 3)$ for $m = 2, ..., 9$ in Shi and Tang (2020).

Table 2.8: Space-filling patterns for subarrays from SOA$(32, 7, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1	1, 4	4
	0, 2, 1	1, 2	4
3	0, 1, 7, 3	1, 4, 7	4
	0, 3, 5, 3	1, 2, 4	4
	0, 5, 3, 3	1, 2, 3	4
	0, 6, 3, 2	1, 2, 7	4
4	0, 9, 19, 11	1, 2, 4, 7	4
	0, 11, 15, 13	1, 2, 3, 6	4
	0, 13, 11, 15	1, 2, 3, 5	4
	1, 8, 14, 16	1, 2, 3, 4	3
5	1, 22, 40, 40	1, 2, 3, 4, 6	3
	1, 24, 34, 46	1, 2, 3, 4, 5	3
6	3, 42, 83, 104	1, 2, 3, 4, 5, 6	3
7	7, 70, 161, 224	1, 2, 3, 4, 5, 6, 7	3

Table 2.9: Space-filling patterns for subarrays from SOA$(32, 8, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1	1, 4	5
	0, 2, 1	1, 2	4
	2, 0, 1	1, 8	4
3	0, 1, 7, 3	1, 4, 6	4
	0, 3, 5, 3	1, 2, 5	4
	0, 5, 3, 3	1, 2, 3	4
	2, 2, 5, 2	1, 4, 5	3
	2, 4, 3, 2	1, 2, 7	3
4	0, 8, 20, 12	1, 2, 3, 5	4
	1, 4, 22, 12	1, 4, 6, 7	3
	1, 8, 14, 16	1, 2, 5, 6	3
	1, 12, 6, 20	1, 2, 3, 4	3
	2, 8, 16, 12	1, 2, 5, 8	3
	2, 10, 12, 14	1, 2, 3, 6	3
	5, 8, 14, 8	1, 4, 5, 8	3
	5, 12, 6, 12	1, 2, 7, 8	3
5	3, 17, 44, 38	1, 2, 3, 5, 8	3
	3, 19, 38, 44	1, 2, 3,5, 6	3
	3, 21, 32, 50	1, 2, 3, 4, 5	3
	5, 20, 34, 38	1, 2, 3, 6, 7	3
	5, 22, 28, 44	1, 2, 3, 6, 8	3
6	7, 36, 79, 108	1, 2, 3, 4, 5, 8	3
	7, 38, 71, 120	1, 2, 3, 4, 5, 6	3
	9, 40, 63, 108	1, 2, 3, 6,7, 8	3
7	13, 62, 143, 248	1, 2, 3, 4, 5, 6, 7	3
8	22, 96, 252, 496	1, 2, 3, 4, 5, 6, 7, 8	3

Table 2.10: Space-filling patterns for subarrays from SOA$(32, 9, 8, 3)$.

m	S_4, S_5, S_6, S_7	Columns	Strength
2	0, 0, 1, -	1, 2	5
	0, 2, 1, -	2, 6	4
	1, 1, 1, -	1, 8	3
	2, 0, 1, -	1, 7	3
3	0, 1, 7, 3	1, 5, 9	4
	0, 1, 8, 1	1, 2, 3	4
	0, 2, 5, 4	1, 2, 5	4
	0, 3, 4, 3	5, 6, 9	4
	0, 3, 5, 3	1, 2, 6	4
	0, 3, 6, 1	2, 5, 9	4
	0, 5, 3, 3	2, 3, 9	4
	0, 6, 3, 2	3, 8, 9	4
	1, 0, 5, 6	1, 4, 5	3
	1, 1, 4, 3	1, 3, 5	3
	1, 1, 7, 1	1, 2, 8	3
	1, 2, 3, 6	1, 3, 9	3
	1, 2, 4, 4	1, 3, 6	3
	1, 3, 5, 1	1, 4, 8	3
	1, 3, 5, 3	1, 2, 9	3
	1, 3, 6, 1	1, 6, 8	3
	2, 1, 4, 3	1, 2, 4	3
	2, 1, 6, 3	1, 3, 4	3
	2, 2, 5, 2	1, 2, 7	3
	2, 3, 4, 3	3, 4, 8	3
	2, 3, 5, 1	3, 4, 6	3

m	S_4, S_5, S_6, S_7	Columns	Strength
3	2, 4, 3, 2	1, 6, 7	3
	4, 2, 4, 2	1, 5, 7	3
4	0, 9, 19, 11	3, 7, 8, 9	4
	1, 5, 23, 9	1, 2, 3, 8	3
	1, 6, 19, 11	1, 5, 6, 9	3
	1, 6, 19, 13	2, 3, 5, 8	3
	1, 7, 15, 19	1, 2, 5, 6	3
	1, 7, 17, 15	5, 6, 8, 9	3
	1, 7, 18, 16	1, 2, 5, 9	3
	1, 8, 14, 18	4, 5, 6, 8	3
	1, 8, 19, 9	1, 5, 6, 8	3
	1, 9, 13, 17	2, 5, 6, 9	3
	1, 10, 13, 15	2, 3, 8, 9	3
	2, 2, 22, 16	1, 4, 5, 6	3
	2, 3, 19, 17	1, 4, 5, 9	3
	2, 4, 16, 20	1, 2, 3, 5	3
	2, 4, 19, 15	1, 3, 5, 6	3
	2, 5, 18, 16	4, 6, 7, 9	3
	2, 5, 20, 8	2, 5, 7, 8	3
	2, 6, 13, 23	2, 5, 8, 9	3
	2, 6, 14, 20	1, 5, 8, 9	3
	2, 6, 15, 17	1, 3, 5, 8	3
	2, 6, 16, 14	1, 2, 3, 6	3
	2, 6, 16, 16	1, 6, 8, 9	3
	2, 6, 18, 12	1, 4, 5, 8	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength	m	S_4, S_5, S_6, S_7	Columns	Strength
4	2, 6, 18, 14	1, 4, 8, 9	3	4	3, 5, 14, 16	1, 3, 5, 9	3
	2, 6, 19, 11	3, 4, 5, 6	3		3, 5, 14, 20	1, 2, 4, 5	3
	2, 7, 13, 21	2, 3, 7, 9	3		3, 5, 14, 20	1, 2, 8, 9	3
	2, 7, 14, 18	1, 2, 3, 9	3		3, 6, 12, 18	3, 4, 5, 9	3
	2, 7, 15, 15	3, 4, 7, 9	3		3, 6, 13, 19	1, 2, 4, 6	3
	2, 7, 15, 17	1, 3, 6, 8	3		3, 6, 13, 19	2, 5, 6, 7	3
	2, 7, 18, 12	1, 2, 6, 8	3		3, 6, 15, 13	2, 4, 5, 7	3
	2, 8, 12, 16	4, 5, 7, 8	3		3, 6, 17, 13	3, 4, 6, 9	3
	2, 8, 13, 19	3, 4, 5, 8	3		3, 6, 18, 10	3, 5, 6, 7	3
	2, 8, 14, 18	3, 5, 7, 9	3		3, 7, 10, 18	2, 6, 7, 8	3
	2, 8, 16, 12	3, 4, 5, 7	3		3, 7, 12, 14	2, 4, 7, 8	3
	2, 9, 9, 23	1, 3, 8, 9	3		3, 7, 14, 14	1, 4, 6, 7	3
	2, 9, 11, 13	1, 2, 6, 9	3		3, 7, 14, 14	3, 6, 7, 9	3
	2, 10, 10, 16	2, 3, 6, 9	3		3, 7, 15, 9	1, 2, 4, 8	3
	2, 10, 10, 18	2, 6, 7, 9	3		3, 7, 17, 11	3, 5, 6, 9	3
	2, 10, 11, 13	2, 4, 6, 7	3		3, 8, 11, 15	1, 3, 6, 7	3
	2, 10, 12, 14	1, 2, 6, 7	3		3, 8, 12, 12	2, 4, 5, 9	3
	2, 10, 13, 15	3, 5, 8, 9	3		3, 8, 13, 17	5, 6, 7, 9	3
	2, 10, 14, 12	3, 6, 8, 9	3		3, 8, 15, 11	1, 2, 4, 9	3
	3, 3, 17, 19	2, 3, 5, 6	3		3, 8, 16, 10	3, 4, 6, 7	3
	3, 3, 18, 18	2, 3, 5, 7	3		3, 10, 13, 11	3, 4, 8, 9	3
	3, 4, 18, 14	1, 2, 3, 7	3		4, 3, 14, 18	1, 3, 4, 5	3
	3, 4, 18, 14	1, 3, 4, 9	3		4, 4, 18, 16	1, 3, 4, 6	3
	3, 5, 11, 21	4, 5, 7, 9	3		4, 5, 10, 20	1, 3, 6, 9	3
	3, 5, 13, 21	1, 3, 7, 9	3		4, 5, 14, 16	1, 3, 4, 8	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength
4	4, 5, 14, 16	1, 3, 7, 8	3
	4, 6, 10, 22	3, 4, 6, 8	3
	4, 6, 13, 19	2, 4, 7, 9	3
	4, 7, 11, 17	1, 6, 7, 9	3
	4, 7, 14, 14	1, 2, 7, 9	3
	4, 7, 15, 11	1, 2, 5, 7	3
	4, 8, 15, 7	1, 4, 7, 8	3
	4, 9, 9, 19	2, 4, 6, 8	3
	4, 9, 10, 12	2, 3, 4, 9	3
	5, 4, 18, 10	1, 2, 3, 4	3
	5, 5, 13, 19	1, 4, 5, 7	3
	5, 5, 14, 16	2, 3, 4, 7	3
	5, 5, 15, 11	1, 2, 7, 8	3
	5, 6, 12, 12	1, 2, 4, 7	3
	5, 6, 12, 14	1, 3, 5, 7	3
	5, 7, 14, 10	1, 5, 6, 7	3
	5, 8, 10, 16	1, 6, 7, 8	3
	5, 9, 11, 9	2, 3, 4, 6	3
	6, 6, 14, 8	1, 5, 7, 8	3
5	3, 16, 42, 46	1, 5, 6, 8, 9	3
	3, 18, 36, 48	1, 2, 5, 6, 9	3
	3, 19, 39, 46	3, 5, 7, 8, 9	3
	4, 10, 49, 44	1, 4, 5, 6, 9	3
	4, 12, 47, 42	2, 3, 5, 7, 8	3
	4, 13, 42, 48	1, 2, 3, 5, 8	3
	4, 13, 44, 48	1, 4, 5, 6, 8	3
	4, 14, 41, 50	1, 3, 5, 6, 8	3
	4, 14, 44, 44	1, 2, 3, 6, 8	3
	4, 15, 36, 60	2, 5, 6, 8, 9	3
	4, 15, 38, 48	4, 5, 6, 8, 9	3
	4, 15, 39, 48	2, 3, 7, 8, 9	3
	4, 16, 36, 56	1, 2, 3, 8, 9	3
	4, 16, 37, 50	1, 2, 5, 6, 8	3
	4, 17, 35, 54	2, 3, 5, 8, 9	3
	4, 17, 37, 48	3, 4, 5, 7, 8	3
	4, 17, 38, 46	3, 6, 7, 8, 9	3
	4, 17, 41, 40	3, 4, 7, 8, 9	3
	4, 19, 33, 50	2, 3, 6, 8, 9	3
	4, 19, 34, 42	2, 4, 5, 6, 9	3
	5, 11, 39, 58	1, 2, 3, 5, 6	3
	5, 11, 41, 58	4, 6, 7, 8, 9	3
	5, 12, 40, 52	1, 4, 5, 8, 9	3
	5, 12, 40, 54	1, 4, 6, 8, 9	3
	5, 13, 35, 62	1, 2, 3, 5, 9	3
	5, 13, 35, 62	1, 2, 5, 8, 9	3
	5, 13, 38, 56	1, 2, 4, 5, 6	3
	5, 14, 34, 58	4, 5, 6, 7, 8	3
	5, 14, 36, 58	2, 3, 5, 7, 9	3
	5, 14, 37, 56	5, 6, 7, 8, 9	3
	5, 14, 38, 48	3, 5, 6, 7, 8	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength	m	S_4, S_5, S_6, S_7	Columns	Strength
5	5, 14, 38, 50	4, 5, 6, 7, 9	3	5	6, 12, 39, 44	1, 3, 5, 6, 9	3
	5, 14, 42, 42	3, 4, 5, 6, 9	3		6, 12, 39, 50	1, 4, 6, 7, 9	3
	5, 15, 33, 58	1, 2, 6, 8, 9	3		6, 13, 36, 56	1, 2, 3, 7, 9	3
	5, 15, 33, 58	3, 4, 5, 6, 8	3		6, 13, 38, 48	1, 3, 4, 7, 9	3
	5, 15, 36, 52	2, 3, 5, 6, 9	3		6, 13, 41, 42	1, 4, 7, 8, 9	3
	5, 15, 37, 50	1, 2, 4, 5, 9	3		6, 14, 29, 58	4, 5, 7, 8, 9	3
	5, 15, 38, 44	1, 2, 3, 6, 7	3		6, 14, 33, 58	3, 4, 6, 7, 8	3
	5, 15, 42, 40	3, 4, 5, 6, 7	3		6, 14, 36, 52	1, 3, 4, 8, 9	3
	5, 16, 30, 60	3, 4, 5, 7, 9	3		6, 14, 37, 46	3, 4, 6, 7, 9	3
	5, 16, 31, 60	1, 3, 5, 8, 9	3		6, 15, 30, 52	2, 6, 7, 8, 9	3
	5, 16, 32, 56	2, 3, 6, 7, 9	3		6, 15, 32, 52	2, 4, 5, 7, 9	3
	5, 16, 34, 54	1, 3, 7, 8, 9	3		6, 15, 34, 52	1, 2, 4, 6, 8	3
	5, 16, 34, 56	2, 5, 6, 7, 9	3		6, 15, 38, 46	1, 2, 5, 7, 9	3
	5, 16, 35, 44	2, 4, 5, 7, 8	3		6, 15, 38, 46	3, 5, 6, 7, 9	3
	5, 16, 35, 48	2, 4, 5, 6, 7	3		6, 16, 27, 52	1, 2, 3, 6, 9	3
	5, 17, 31, 56	1, 3, 6, 8, 9	3		6, 16, 31, 54	1, 3, 6, 7, 8	3
	5, 17, 33, 46	1, 2, 4, 6, 9	3		6, 16, 34, 48	1, 2, 4, 8, 9	3
	5, 18, 30, 56	2, 4, 5, 6, 8	3		6, 16, 34, 50	3, 4, 6, 8, 9	3
	5, 18, 31, 52	2, 4, 6, 7, 9	3		6, 16, 35, 48	2, 3, 4, 5, 8	3
	5, 18, 38, 40	3, 5, 6, 8, 9	3		6, 17, 26, 56	2, 4, 6, 7, 8	3
	6, 9, 45, 50	2, 3, 5, 6, 7	3		6, 17, 27, 56	2, 4, 5, 8, 9	3
	6, 9, 46, 48	1, 3, 4, 5, 6	3		6, 17, 29, 54	3, 4, 5, 8, 9	3
	6, 11, 39, 54	1, 3, 4, 5, 8	3		6, 17, 30, 56	2, 4, 6, 8, 9	3
	6, 11, 44, 42	1, 2, 3, 7, 8	3		6, 17, 32, 48	1, 2, 5, 6, 7	3
	6, 12, 35, 56	2, 5, 6, 7, 8	3		6, 18, 29, 46	1, 2, 4, 6, 7	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength	m	S_4, S_5, S_6, S_7	Columns	Strength
5	6, 19, 26, 50	1, 2, 6, 7, 9	3	5	7, 16, 35, 44	1, 3, 4, 6, 7	3
	6, 19, 29, 44	2, 3, 4, 6, 9	3		7, 17, 34, 40	2, 3, 4, 6, 7	3
	7, 9, 41, 52	1, 3, 4, 6, 9	3		7, 20, 28, 42	2, 3, 4, 8, 9	3
	7, 10, 36, 50	1, 3, 4, 5, 9	3		8, 10, 35, 56	1, 2, 3, 4, 5	3
	7, 11, 35, 56	1, 4, 5, 7, 9	3		8, 12, 33, 50	1, 2, 7, 8, 9	3
	7, 11, 36, 60	1, 3, 4, 6, 8	3		8, 13, 37, 44	1, 2, 3, 4, 6	3
	7, 12, 36, 52	1, 2, 3, 5, 7	3		8, 13, 39, 34	1, 2, 5, 7, 8	3
	7, 12, 40, 46	1, 4, 5, 6, 7	3		8, 14, 32, 46	1, 3, 4, 5, 7	3
	7, 13, 32, 56	1, 3, 5, 7, 9	3		8, 14, 36, 40	1, 4, 5, 7, 8	3
	7, 13, 33, 54	1, 5, 7, 8, 9	3		8, 15, 31, 48	1, 2, 4, 5, 7	3
	7, 13, 37, 46	2, 3, 4, 7, 8	3		8, 15, 33, 42	1, 5, 6, 7, 8	3
	7, 13, 37, 48	2, 3, 4, 5, 7	3		8, 15, 34, 42	1, 2, 4, 7, 9	3
	7, 13, 41, 34	1, 2, 3, 4, 8	3		8, 16, 28, 52	2, 3, 4, 6, 8	3
	7, 14, 28, 58	1, 3, 6, 7, 9	3		8, 16, 32, 34	1, 2, 4, 7, 8	3
	7, 14, 31, 54	2, 4, 7, 8, 9	3		9, 13, 36, 38	1, 2, 3, 4, 7	3
	7, 14, 32, 56	1, 6, 7, 8, 9	3	6	8, 26, 89, 121	1, 4, 5, 6, 8, 9	3
	7, 14, 35, 48	2, 3, 4, 5, 6	3		8, 29, 83, 124	2, 3, 5, 7, 8, 9	3
	7, 14, 36, 42	1, 2, 3, 4, 9	3		8, 30, 75, 139	1, 2, 5, 6, 8, 9	3
	7, 14, 36, 44	1, 3, 5, 6, 7	3		8, 31, 79, 118	1, 2, 4, 5, 6, 9	3
	7, 15, 28, 56	2, 3, 4, 5, 9	3		9, 26, 82, 127	1, 2, 3, 5, 6, 8	3
	7, 15, 30, 56	2, 3, 4, 7, 9	3		9, 28, 73, 148	1, 2, 3, 5, 8, 9	3
	7, 15, 31, 52	1, 4, 6, 7, 8	3		9, 29, 78, 128	1, 2, 4, 5, 6, 8	3
	7, 15, 35, 42	1, 3, 4, 7, 8	3		9, 30, 74, 127	2, 3, 6, 7, 8, 9	3
	7, 16, 31, 46	1, 2, 6, 7, 8	3		9, 30, 77, 128	2, 3, 5, 6, 8, 9	3
	7, 16, 34, 44	1, 5, 6, 7, 9	3		9, 30, 78, 123	1, 3, 5, 6, 8, 9	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength	m	S_4, S_5, S_6, S_7	Columns	Strength
6	9, 31, 70, 136	1, 2, 3, 6, 8, 9	3	6	10, 33, 69, 122	1, 2, 5, 6, 7, 9	3
	9, 31, 80, 116	3, 5, 6, 7, 8, 9	3		11, 21, 89, 115	1, 3, 4, 5, 6, 9	3
	9, 32, 70, 131	2, 4, 5, 6, 8, 9	3		11, 25, 75, 135	1, 3, 4, 5, 8, 9	3
	9, 32, 70, 131	3, 4, 5, 7, 8, 9	3		11, 25, 77, 135	1, 4, 6, 7, 8, 9	3
	9, 32, 72, 125	2, 4, 5, 6, 7, 9	3		11, 25, 77, 137	1, 3, 4, 6, 8, 9	3
	10, 22, 87, 127	2, 3, 5, 6, 7, 8	3		11, 25, 79, 127	1, 2, 3, 5, 6, 7	3
	10, 23, 85, 136	1, 3, 4, 5, 6, 8	3		11, 25, 81, 119	1, 2, 3, 4, 5, 8	3
	10, 26, 74, 140	4, 5, 6, 7, 8, 9	3		11, 25, 82, 120	1, 4, 5, 6, 7, 9	3
	10, 26, 76, 138	2, 5, 6, 7, 8, 9	3		11, 25, 84, 112	1, 2, 3, 5, 7, 8	3
	10, 26, 80, 134	2, 3, 5, 6, 7, 9	3		11, 26, 73, 142	1, 2, 3, 5, 7, 9	3
	10, 27, 72, 135	1, 2, 3, 5, 6, 9	3		11, 27, 81, 115	2, 3, 4, 5, 6, 7	3
	10, 27, 79, 126	1, 2, 3, 7, 8, 9	3		11, 28, 67, 134	2, 4, 5, 7, 8, 9	3
	10, 27, 81, 118	1, 2, 3, 6, 7, 8	3		11, 28, 75, 120	1, 3, 5, 6, 7, 8	3
	10, 28, 73, 133	3, 4, 5, 6, 7, 8	3		11, 29, 65, 143	2, 3, 4, 5, 7, 9	3
	10, 28, 74, 132	1, 2, 4, 5, 8, 9	3		11, 29, 65, 135	1, 2, 3, 6, 7, 9	3
	10, 28, 76, 130	3, 4, 6, 7, 8, 9	3		11, 29, 65, 139	2, 4, 6, 7, 8, 9	3
	10, 28, 78, 124	3, 4, 5, 6, 7, 9	3		11, 29, 69, 133	2, 3, 4, 5, 6, 8	3
	10, 28, 80, 118	2, 3, 4, 5, 7, 8	3		11, 29, 73, 127	1, 5, 6, 7, 8, 9	3
	10, 29, 69, 136	2, 4, 5, 6, 7, 8	3		11, 30, 71, 122	1, 2, 4, 5, 6, 7	3
	10, 29, 79, 116	1, 3, 4, 7, 8, 9	3		11, 30, 72, 119	2, 3, 4, 7, 8, 9	3
	10, 30, 70, 134	1, 2, 4, 6, 8, 9	3		11, 30, 74, 115	1, 2, 3, 4, 8, 9	3
	10, 30, 72, 132	1, 3, 5, 7, 8, 9	3		11, 31, 67, 127	2, 3, 4, 6, 7, 9	3
	10, 30, 74, 124	3, 4, 5, 6, 8, 9	3		11, 32, 64, 131	2, 3, 4, 5, 8, 9	3
	10, 31, 68, 135	1, 3, 6, 7, 8, 9	3		11, 32, 66, 119	1, 2, 4, 6, 7, 9	3
	10, 31, 72, 121	2, 3, 4, 5, 6, 9	3		11, 34, 63, 126	2, 3, 4, 6, 8, 9	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength
6	12, 23, 79, 132	1, 2, 3, 4, 5, 6	3
	12, 25, 70, 137	1, 2, 3, 4, 5, 9	3
	12, 25, 74, 127	1, 4, 5, 7, 8, 9	3
	12, 25, 75, 128	1, 2, 5, 7, 8, 9	3
	12, 26, 68, 134	1, 3, 4, 5, 7, 9	3
	12, 26, 76, 122	1, 4, 5, 6, 7, 8	3
	12, 26, 77, 123	1, 2, 3, 4, 6, 8	3
	12, 26, 80, 114	1, 3, 4, 5, 6, 7	3
	12, 27, 73, 122	1, 3, 5, 6, 7, 9	3
	12, 27, 75, 118	1, 3, 4, 5, 7, 8	3
	12, 28, 68, 132	2, 3, 4, 6, 7, 8	3
	12, 28, 69, 133	1, 3, 4, 6, 7, 8	3
	12, 28, 70, 118	1, 2, 3, 4, 6, 9	3
	12, 28, 71, 119	1, 2, 5, 6, 7, 8	3
	12, 29, 62, 135	1, 2, 6, 7, 8, 9	3
	12, 29, 70, 123	1, 2, 4, 5, 7, 9	3
	12, 31, 63, 122	1, 2, 4, 6, 7, 8	3
	13, 27, 72, 118	1, 2, 3, 4, 7, 9	3
	13, 27, 77, 99	1, 2, 3, 4, 7, 8	3
	13, 28, 70, 113	1, 2, 4, 7, 8, 9	3
	13, 29, 73, 103	1, 2, 4, 5, 7, 8	3
	13, 30, 72, 107	1, 2, 3, 4, 6, 7	3
	14, 26, 69, 121	1, 2, 3, 4, 5, 7	3
7	15, 52, 145, 278	1, 2, 4, 5, 6, 8, 9	3
	16, 48, 152, 274	2, 3, 5, 6, 7, 8, 9	3

m	S_4, S_5, S_6, S_7	Columns	Strength
7	16, 50, 140, 292	1, 2, 3, 5, 6, 8, 9	3
	17, 45, 154, 275	1, 3, 4, 5, 6, 8, 9	3
	17, 48, 145, 284	1, 2, 3, 5, 7, 8, 9	3
	17, 50, 134, 292	2, 4, 5, 6, 7, 8, 9	3
	17, 50, 139, 282	5, 4, 5, 6, 7, 8, 9	3
	17, 51, 135, 283	1, 2, 3, 6, 7, 8, 9	3
	17, 51, 140, 277	2, 3, 4, 5, 6, 7, 9	3
	17, 52, 141, 268	1, 3, 5, 6, 7, 8, 9	3
	17, 52, 135, 280	2, 3, 4, 5, 7, 8, 9	3
	17, 54, 133, 276	2, 3, 4, 5, 6, 8, 9	3
	17, 54, 137, 260	1, 2, 4, 5, 6, 7, 9	3
	18, 45, 148, 279	1, 2, 3, 4, 5, 6, 8	3
	18, 45, 150, 265	1, 2, 3, 5, 6, 7, 8	3
	18, 46, 146, 278	1, 4, 5, 6, 7, 8, 9	3
	18, 47, 142, 273	1, 2, 3, 4, 5, 6, 9	3
	18, 47, 142, 279	2, 3, 4, 5, 6, 7, 8	3
	18, 48, 138, 286	1, 2, 3, 4, 5, 8, 9	3
	18, 48, 140, 286	1, 3, 4, 6, 7, 8, 9	3
	18, 49, 138, 277	1, 3, 4, 5, 7, 8, 9	3
	18, 50, 134, 284	1, 2, 5, 6, 7, 8, 9	3
	18, 51, 135, 277	1, 2, 3, 4, 6, 8, 9	3
	18, 52, 130, 282	2, 3, 4, 6, 7, 8, 9	3
	19, 45, 146, 267	1, 3, 4, 5, 6, 7, 9	3
	19, 46, 143, 274	1, 3, 4, 5, 6, 7, 8	3
	19, 50, 135, 266	1, 2, 4, 5, 6, 7, 8	3

Table 2.10: Continued.

m	S_4, S_5, S_6, S_7	Columns	Strength
7	19, 50, 141, 248	1, 2, 3, 4, 7, 8, 9	3
	19, 51, 126, 281	1, 2, 4, 6, 7, 8, 9	3
	20, 47, 142, 261	1, 2, 3, 4, 5, 6, 7	3
	20, 47, 144, 247	1, 2, 3, 4, 5, 7, 8	3
	20, 48, 134, 268	1, 2, 4, 5, 7, 8, 9	3
	20, 48, 128, 288	1, 2, 3, 4, 5, 7, 9	3
	20, 49, 136, 261	1, 2, 3, 4, 6, 7, 8	3
	20, 50, 130, 266	1, 2, 3, 4, 6, 7, 9	3
8	27, 80, 248, 546	1, 2, 3, 5, 6, 7, 8, 9	3
	27, 82, 242, 548	2, 3, 4, 5, 6, 7, 8, 9	3
	28, 78, 250, 540	1, 3, 4, 5, 6, 7, 8, 9	3
	28, 82, 238, 540	1, 2, 4, 5, 6, 7, 8, 9	3
	29, 80, 240, 534	1, 2, 3, 4, 5, 6, 7, 9	3
	29, 82, 234, 536	1, 2, 3, 4, 6, 7, 8, 9	3
	30, 76, 248, 524	1, 2, 3, 4, 5, 6, 7, 8	3
9	42, 124, 400, 976	1, 2, 3, 4, 5, 6, 7, 8, 9	3

CHAPTER 3

A Stratified L_2-Discrepancy for Uniform Designs

In this chapter, we propose a new discrepancy, the stratified L_2-discrepancy to measure the uniformity of designs. The new discrepancy evaluates design stratification properties on various kinds of grids. The idea of stratification comes from the stratified orthogonality of strong orthogonal arrays and is implemented on a continuous space. The stratified L_2-discrepancy is applicable to all designs on the unit hypercube. Using the framework of reproducing kernel, the proposed discrepancy inherits most of the advantages of the generalized L_2-discrepancy. The nature of stratification avoids design points gathering around certain locations for a high-dimensional design space. Designs with small stratified L_2-discrepancy are space-filling and have good projection properties. The stratified L_2-discrepancy is general and flexible in measuring projection properties with associated stratification schemes and importance levels. In particular, the space-filling criterion proposed by Tian and Xu (2021) is a special case of the stratified L_2-discrepancy when the space-filling hierarchy principle is assumed.

3.1 Notation and Backgrounds

The generalized L_2-discrepancy was proposed by Hickernell (1998) to overcome the shortcomings of the star L_p-discrepancy. Let $\mathcal{P}$ be a design with n runs and m factors on the unit hypercube $C^m = [0,1)^m = [0,1) \times \cdots \times [0,1)$. Let $\{1:m\} = \{1,\ldots,m\}$ and let $u \subseteq \{1:m\}$ be the set used to index the factors of interest. For each $x \in C^m$, let $R_u(x) \subseteq [0,1)^u$ be a neighboring region of x in C^m in the coordinates indexed by u. Define the local projection discrepancy for the factors indexed by u as

$$\mathrm{disc}_u^R(x) = \mathrm{Vol}(R_u(x)) - \frac{|\mathcal{P} \cap R_u(x)|}{n}, \tag{3.1}$$

where $|\mathcal{P} \cap R_u(x)|$ is the number of design points in $\mathcal{P}$ that lie inside the region $R_u(x)$ in the coordinates indexed by u and $\mathrm{Vol}(R_u(x))$ is the volume of $R_u(x)$. The local projection discrepancy is a function of x and u that measures the difference between the volume of the region $R_u(x)$ and the proportion of design points that fall into the region.

A generalized L_2-discrepancy is defined in terms of all local projection discrepancies as follows:

$$D_2^R(\mathcal{P}) = \left\{ \int_{[0,1]^m} \sum_{u \subseteq \{1:m\}} |\mathrm{disc}_u^R(x_u)|^2 dx \right\}^{\frac{1}{2}}.$$

The generalized L_2-discrepancy considers local projection discrepancies for all u with equal weights. Projection properties of the design are evaluated for each set of sub-dimensions. The definition of $R_u(x)$ for the generalized L_2-discrepancy makes it possible to design the neighboring region. The centered L_2-discrepancy defines $R_u(x)$ to be the hyperrectangle between x and the nearest vertex and the wrap-around L_2-discrepancy defines the region $R_u(x, y)$ to be the hyperrectangle region between x and y wrapping the unit cube. If the volume of the region is relevant to x, it may cause undesirable problems. For example, because the region $R_u(x)$ for the centered L_2-discrepancy involves vertexes of the design region, designs selected by the centered L_2-discrepancy favor design points around the center point in a high-dimensional input space. To overcome these issues, we define the stratified L_2-discrepancy with a stratified region $R_u(x)$ so that projection properties are evaluated based on design stratifications on various grids flexibly.

3.2 A stratified L_2-Discrepancy

3.2.1 A stratified region $R_u(x)$

For any x in $[0,1)$ and given integers s and p, we define a series of mapping functions $f_k(x)$ for $k \geq 1$ as follows:

$$f_k(x) = \lfloor \lfloor s^p x \rfloor / s^{p-k} \rfloor \mod s,$$

where $\lfloor x \rfloor$ denotes the largest integer not exceeding x. The functions $f_k(x)$ gives out the kth digit of x after the radix point in base s numeral system, that is, $x = \sum_{k=1}^{\infty} f_k(x) s^{-k}$. For convenience, we express x as $(0.f_1 f_2 \dots f_p \dots)_s$ in base s numeral system. For example, when $s = 2$, we can write $3/8$ as $(0.011)_2$, $7/8$ as $(0.111)_2$, and $7/16$ as $(0.0111)_2$. The first p digits after the radix point are sufficient to determine the position of x when $[0,1)$ is evenly divided into s^p intervals.

Let $\mathbb{Z}_{p+1} = \{0, 1, \cdots, p\}$ be a set of integers. For $u = (u_1, \cdots, u_m) \in \mathbb{Z}_{p+1}^m$ and $x = (x_1, \dots, x_m) \in [0,1)^m$, the stratified region $R_u(x)$ is defined by

$$R_u(x) = \bigotimes_{j=1}^{m} R_{u_j}(x_j), \tag{3.2}$$

where $\bigotimes$ is the Kronecker product, $R_{u_j}(x_j) = \left[\sum_{k=1}^{u_j} f_k(x_j) s^{-k}, \sum_{k=1}^{u_j} f_k(x_j) s^{-k} + s^{-u_j}\right)$ for $u_j = 1, \cdots, p$ and $R_0(x_j) = [0,1)$. The interval $R_{u_j}(x_j)$ contains x_j and has length s^{-u_j}; it is uniquely determined by the first u_j digits of x_j after the radix point in base s numeral system.

The region $R_u(x)$ is a hyperrectangle based on u and x. The value of u decides how many grids are drawn in the design region and x decides on which grid $R_u(x)$ represents. In the jth dimension, the design region is divided to s^{u_j} grids. Because the length of the grid on the jth dimension is s^{-u_j}, the volume of $R_u(x)$ is $\Pi_{j=1}^m s^{-u_j}$. For any x and a fixed u, there is one and only one region $R_u(x)$ that contains x.

Here we present a simple example to illustrate the region $R_u(x)$. Consider a two-dimensional design space and $s = 2, p = 3$. We randomly draw points x in $[0,1)^2$ and represent points as triangles. In Figure 3.1, the boxes are the regions $R_u(x)$ containing randomly drawn points x with

u specified on the top of the plot. For instance, in the first plot in the first row, the box shows the region $R_u(x)$ when $u = (0, 1)$ and $x = (9/16, 11/16)$. Here $u_1 = 0, u_2 = 1$ means that the design region will be divided into 2^0 part and 2^1 parts in the first and second dimension, respectively. The box $R_u(x)$ shows a rectangle that contains x where the horizontal and vertical axes represent the first and second dimensions. When $u_1 = u_2 = 1$ in the third plot in the first row, the design region in both dimensions are divided into 2 parts and $R_u(x)$ is the square box that contains x. The finest grids we can draw are $2^3 \times 2^3$ grids when $u = (3, 3)$. For plots with the same $u_1 + u_2$, the volumes of the boxes are the same.

The stratified region $R_u(x)$ defines multiple kinds of neighboring grids around x based on u. The index set u determines the shape and the volume of the grid. Projection properties can be assessed based on all $u \in \mathbb{Z}^m_{p+1}$ in a flexible way. The volume of $R_u(x)$ is the same for different x as long as u is fixed; consequently points are treated equally important regardless of its location. In addition, the values of s and p provide flexibility in the definition of the stratified region.

3.2.2 A stratified L_2-discrepancy

Let $\mathcal{P}$ be a design with n runs and m factors on C^m. The stratified L_2-discrepancy is defined with the local projection discrepancy in (3.1) given the stratified region $R_u(x)$ in (3.2). The projection property is considered by summing over all local projection discrepancies for $u \in \mathbb{Z}^m_{p+1}$. We further introduce weights related to u in the formulation of the stratified L_2-discrepancy to allocate different importance level for various stratifications. The definition of the stratified L_2-discrepancy is

$$SD_2^R(\mathcal{P}) = \left\{ \int_{[0,1]^m} \sum_{u \in \mathbb{Z}^m_{p+1}} w(u) |\mathrm{disc}_u^R(x)|^2 dx \right\}^{\frac{1}{2}}, \tag{3.3}$$

where weight $w(u) = \prod_{i=1}^m w(u_i) > 0$ is the product of individual weights of each u_i. When $w(u) = 1, \forall u \in \mathbb{Z}^m_{p+1}$, the projection uniformities of all stratifications are treated equally important. If we believe in the space-filling hierarchy principle, the weight $w(u)$ should be defined in terms of the size of the grids in the stratifications. The stratified L_2-discrepancy can be tuned to

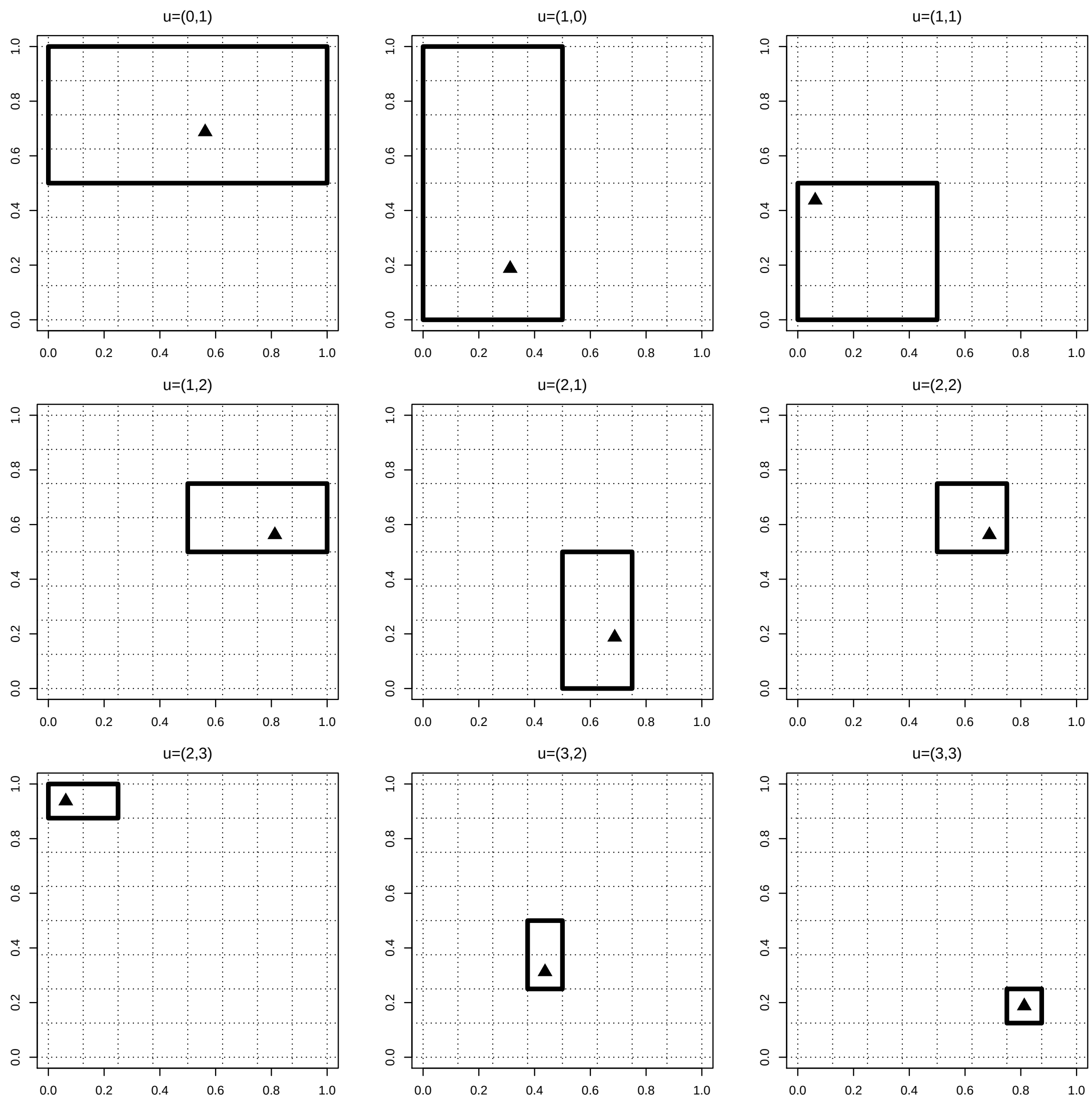

Figure 3.1: Illustration of the stratified region $R_u(x)$.

emphasize projection properties flexibly. Note that $R_u(x)$ in (3.2) is only defined for $x \in [0,1)^m$ whereas the integration in (3.3) is over $x \in [0,1]^m$. This does not cause a problem because the integration leads to the same value whether it is over $[0,1)^m$ or $[0,1]^m$.

To derive an analytical expression for the stratified L_2-discrepancy, we need to introduce the concept of a reproducing kernel Hilbert space. Let $\mathcal{X}$ be a design space. A kernel function $\mathcal{K}(t,z)$ on $\mathcal{X}^2 = \mathcal{X} \times \mathcal{X}$ is symmetric and nonnegative definite. Let $\mathcal{W}$ be the real-valued function space on $\mathcal{X}$ with the kernel function $\mathcal{K}$ as follows (Fang et al., 2018):

$$\mathcal{W} = \big\{F(x) : \int_{\mathcal{X}^2} \mathcal{K}(t,z)dF(t)dF(z) < \infty\big\}.$$

The inner product between two functions $F, G \in \mathcal{W}$ is defined as

$$\langle F, G\rangle_{\mathcal{W}} = \int_{\mathcal{X}^2} \mathcal{K}(t,z)dF(t)dG(z).$$

The norm of the function $F \in \mathcal{W}$ is $\|F\|_{\mathcal{W}} = [\langle F, F\rangle_{\mathcal{W}}]^{\frac{1}{2}}$. Then $\mathcal{W}$ is a Hilbert space of real-valued functions on $\mathcal{X}$. If the kernel function is a reproducing kernel, then $\mathcal{W}$ is called a reproducing kernel Hilbert space.

Now we define the kernel function $\mathcal{K}(t,z)$ based on the stratified region $R_u(x)$ in (3.2) on $\mathcal{X}^2 = C^m \times C^m$ as follows:

$$\mathcal{K}^R(t,z) = \sum_{u \in \mathbb{Z}^m_{p+1}} \mathcal{K}^R_u(t,z), \tag{3.4}$$

where

$$\mathcal{K}^R_u(t,z) = \int_{C^m} w(u) 1_{R_u(x)}(t) 1_{R_u(x)}(z) dx$$

and $1_{R_u(x)}(t) = 1$ if $t \in R_u(x)$ and 0 otherwise. The next theorem establishes the connection between the stratified L_2-discrepancy and the reproducing Hilbert space $\mathcal{W}$.

Theorem 3.1. *Let F be the cumulative distribution function for the uniform distribution on C^m and let $F_{\mathcal{P}}$ be the empirical distribution function for design $\mathcal{P}$ on C^m. Then the stratified L_2-discrepancy of design $\mathcal{P}$ can be defined in the form of a norm in a reproducing Hilbert space $\mathcal{W}$ with kernel $\mathcal{K}^R(t,z)$ as*

$$SD^R_2(\mathcal{P}) = \|F - F_{\mathcal{P}}\|_{\mathcal{W}}. \tag{3.5}$$

With Theorem 3.1, the calculation of the stratified L_2-discrepancy becomes the integral of the kernel function with respect to distribution functions, from which simple analytical expression can be derived.

The kernel function $\mathcal{K}^R(t,z)$ in (3.4) is the sum of $\mathcal{K}_u^R(t,z)$ over all possible u. For $t = (t_1,\ldots,t_m)$ and $z = (z_1,\ldots,z_m)$ in $[0,1)^m$, each $\mathcal{K}_u^R(t,z)$ is a separable kernel function and equals to the product of kernel functions in each dimension: $\mathcal{K}_u^R(t,z) = \prod_{j=1}^m \tilde{\mathcal{K}}_{u_j}^R(t_j,z_j)$. In the jth dimension, $\tilde{\mathcal{K}}_{u_j}^R(t_j,z_j)$ can be simplified as

$$\tilde{\mathcal{K}}_{u_j}^R(t_j,z_j) = \int_0^1 w(u_j)1_{R_{u_j}(x_j)}(t_j)1_{R_{u_j}(x_j)}(z_j)dx_j = w(u_j)s^{-u_j}\delta_{u_j}(t_j,z_j)$$

where

$$\delta_i(t_j,z_j) = \prod_{k=1}^{i} 1_{f_k(t_j)=f_k(z_j)} \tag{3.6}$$

for $i = 1,\cdots,p$ are the indicator functions revealing whether t_j and z_j are in the same grid when $[0,1)$ is stratified to s^i parts. We define $w(0) = 1$ and $\delta_0(t_j,z_j) = 1$ to make the formula consistent. In the jth dimension, the kernel function $\tilde{\mathcal{K}}_{u_j}^R(t_j,z_j)$ takes value $w(u_j)\text{Vol}(R_{u_j}(x))$ or 0 depending on the relative locations of t_j and z_j. When t_j and z_j lie in the same grid, the kernel function takes the larger value as a penalty.

By the binomial theorem, the kernel $\mathcal{K}^R(t,z)$ can be expressed in a product form:

$$\mathcal{K}^R(t,z) = \prod_{j=1}^m \left[1 + \sum_{i=1}^p \tilde{\mathcal{K}}_i^R(t_j,z_j)\right].$$

Thus, tedious summation over all possible $u \in \mathbb{Z}_{p+1}^m$ is avoided. In the jth dimension, all possible stratifications are considered in the form of $\sum_{i=1}^p \tilde{\mathcal{K}}_i^R(t_j,z_j)$. The kernel function of the stratified L_2-discrepancy consists of terms reflecting the uniformity of all possible stratifications of the design without adding too much complexity to the computation. As a result, we can derive simple and fast analytical expressions that can efficiently compute the stratified L_2-discrepancy.

Example 3.1. *When $s = 2, p = 3$, the kernel function of the stratified L_2-discrepancy is*

$$\mathcal{K}^R(t,z) = \prod_{j=1}^m [1 + \frac{w(1)}{2}\delta_1(t_j,z_j) + \frac{w(2)}{4}\delta_2(t_j,z_j) + \frac{w(3)}{8}\delta_3(t_j,z_j)].$$

Suppose that $m = 1$ and all weights are equal to 1. The values of $\mathcal{K}^R(t,z)$ are displayed in Table 3.1 for t, z in each of the grids when $[0,1)$ is stratified to 8 parts equally. If t and z are not in the

Table 3.1: The kernel function $\mathcal{K}^R(t,z)$ for $m=1$, $s=2$ and $p=3$.

	Region of z							
Region of t	$[0,1/8)$	$[1/8,1/4)$	$[1/4,3/8)$	$[3/8,1/2)$	$[1/2,5/8)$	$[5/8,3/4)$	$[3/4,7/8)$	$[7/8,1)$
$[0,1/8)$	1.875	1.75	1.5	1.5	1	1	1	1
$[1/8,1/4)$	1.75	1.875	1.5	1.5	1	1	1	1
$[1/4,3/8)$	1.5	1.5	1.875	1.75	1	1	1	1
$[3/8,1/2)$	1.5	1.5	1.75	1.875	1	1	1	1
$[1/2,5/8)$	1	1	1	1	1.875	1.75	1.5	1.5
$[5/8,3/4)$	1	1	1	1	1.75	1.875	1.5	1.5
$[3/4,7/8)$	1	1	1	1	1.5	1.5	1.875	1.75
$[7/8,1)$	1	1	1	1	1.5	1.5	1.75	1.875

same half, $\mathcal{K}^R(t,z)=1$. If t and z are in the same half but not in the same quarter, $\mathcal{K}^R(t,z)=1.5$. If t and z are in the same quarter but not in the same one eighth, $\mathcal{K}^R(t,z)=1.75$. If t and z are in the same one eighth, $\mathcal{K}^R(t,z)=1.875$. Large kernel values lead to large discrepancies. The stratified L_2-discrepancy discourages designs with points close to each other with $p+1$ stair-wise penalties based on the value of s.

Example 3.2. *When $s=3,p=2$, the kernel function of the stratified L_2-discrepancy is*

$$\mathcal{K}^R(t,z)=\prod_{j=1}^{m}[1+\frac{w(1)}{3}\delta_1(t_j,z_j)+\frac{w(2)}{9}\delta_2(t_j,z_j)].$$

We present the values of $\mathcal{K}^R(t,z)$ in Table 3.2 for t,z in each of the grids when $[0,1)$ is stratified to 9 parts equally. The kernel function takes three possible values: $1,1.33,1.44$ depending on the relative locations of t and z. If t and z are not in the same one third of the design region, $\mathcal{K}^R(t,z)=1$. If t and z are in the same one third but not in the same one ninth, $\mathcal{K}^R(t,z)=1.33$. If t and z are in the same one ninth, $\mathcal{K}^R(t,z)=1.44$.

In the above examples, we utilize equal weights for all projections. The choice of s,p and weights provides flexibility to the stratified L_2-discrepancy and enables a deeper investigation to

Table 3.2: The kernel function $\mathcal{K}^R(t, z)$ for $m = 1$, $s = 3$ and $p = 2$.

	Region of z								
Region of t	$[0, 1/9)$	$[1/9, 2/9)$	$[2/9, 1/3)$	$[1/3, 4/9)$	$[4/9, 5/9)$	$[5/9, 2/3)$	$[2/3, 7/9)$	$[7/9, 8/9)$	$[8/9, 1)$
$[0, 1/9)$	1.44	1.33	1.33	1	1	1	1	1	1
$[1/9, 2/9)$	1.33	1.44	1.33	1	1	1	1	1	1
$[2/9, 1/3)$	1.33	1.33	1.44	1	1	1	1	1	1
$[1/3, 4/9)$	1	1	1	1.44	1.33	1.33	1	1	1
$[4/9, 5/9)$	1	1	1	1.33	1.44	1.33	1	1	1
$[5/9, 2/3)$	1	1	1	1.33	1.33	1.44	1	1	1
$[2/3, 7/9)$	1	1	1	1	1	1	1.44	1.33	1.33
$[7/9, 8/9)$	1	1	1	1	1	1	1.33	1.44	1.33
$[8/9, 1)$	1	1	1	1	1	1	1.33	1.33	1.44

the projection properties of the designs. The choice of these parameters also allows implementation of prior knowledge and preference to the criterion. We will address this in Section 3.4.2.

In the rest of this section, we provide the analytic expression for the stratified L_2-discrepancy. For design $\mathcal{P} = \{x_1, \cdots, x_n\} \subset C^m$, the L_2-discrepancy in (3.5) has the following computational formula according to (2.4.5) of Fang et al. (2018) :

$$\begin{aligned} SD_2^R(\mathcal{P})^2 &= \int_{C^{2m}} \mathcal{K}^R(t, z) d(F - F_{\mathcal{P}})(t) d(F - F_{\mathcal{P}})(z) \\ &= \int_{C^{2m}} \mathcal{K}^R(t, z) dF(t) dF(z) - \frac{2}{n} \sum_{a=1}^{n} \int_{C^m} \mathcal{K}^R(t, x_a) dF(t) + \frac{1}{n^2} \sum_{a,b=1}^{n} \mathcal{K}^R(x_a, x_b). \end{aligned}$$

This provides us an analytical expression for the stratified L_2-discrepancy.

Theorem 3.2. *For design $\mathcal{P} = \{x_1, \cdots, x_n\} \subset C^m$, $x_i = (x_{i1}, \cdots, x_{im}), i = 1, \cdots, n$, the stratified L_2-discrepancy has the following expression:*

$$SD_2^R(\mathcal{P})^2 = -\left(1 + \sum_{i=1}^{p} w(i) s^{-2i}\right)^m + \frac{1}{n^2} \sum_{a,b=1}^{n} \prod_{j=1}^{m} \left[1 + \sum_{i=1}^{p} w(i) s^{-i} \delta_i(x_{aj}, x_{bj})\right], \quad (3.7)$$

where $\delta_i(\cdot, \cdot)$ is an indicator function defined in (3.6).

Given a design with n runs and m columns over C^m, the stratified L_2-discrepancy depends on the choices of s, p and $w(i), i = 1, \ldots, p$. Here are some rules of thumb. Constant weights $w(i) = 1$ work well enough in most of the situations when m is not too large. We use constant weights in the stratified L_2-discrepancy unless specially mentioned for the rest of the chapter. A good choice for s is a small number such as 2 or 3. We choose p to be the largest p such that $s^p \leq n$.

3.3 Applications

The implementation of the stratified L_2-discrepancy is straightforward. We present some examples to demonstrate our newly proposed discrepancy. For a q-level design $D = (d_{ij})$ with entries from $\mathbb{Z}_q$, its stratified L_2-discrepancy $SD_2^R(D)$ is defined to be $SD_2^R(\mathcal{P})$ where $\mathcal{P} = (x_{ij})$ with $x_{ij} = (d_{ij} + 0.5)/q$.

Example 3.3. *Consider four 19×18 Latin hypercube designs studied by Sun et al. (2019). The four designs consist of a uniform design under the centered L_2-discrepancy, a maximin distance design, a maximum projection design and a uniform projection design. The uniform projection design has good projection properties and is robust under various criteria (Sun et al., 2019). We calculate the stratified L_2-discrepancy (SD) with $s = 2, 3$ and other discrepancies for the four designs. The results are in Table 3.3. The stratified L_2-discrepancy prefers the uniform projection design. The centered L_2-discrepancy (CD) and the mixture L_2-discrepancy (MD) choose the uniform design, while the warp-around L_2-discrepancy (WD) chooses the maximum projection design. The stratified L_2-discrepancy confirms that the uniform projection design has good projection properties.*

Example 3.4. *We randomly generate four types of designs: orthogonal array-based samples (OAS), orthogonal array-based Latin hypercube samples (OALHS), Latin hypercube samples (LHS) and maximin Latin hypercube samples (mLHS) and evaluate them with different discrepancies. The designs have 49 runs and 8 columns. Each type of designs are generated for 20 times using the R package `lhs`. Figure 3.2 shows the values of the discrepancies. The stratified L_2-discrepancy*

Table 3.3: Discrepancies for four 19×18 Latin hypercube designs.

Design	CD	WD	MD	SD ($s = 2$)	SD ($s = 3$)
Maximin	1.2889	7.0488	25.2549	87.7170	6.0710
MaxPro	1.3090	**6.8823**	24.8515	87.6938	6.0468
Uniform	**1.2643**	6.9414	**24.8049**	87.6903	6.0496
UniformPro	1.2655	6.9352	24.8554	**87.6342**	**6.0365**

picks the same top two designs as the centered L_2-discrepancy and the mixture L_2-discrepancy. The orthogonal array-based Latin hypercube samples are the most space-filling followed by the maximin Latin hypercube samples. For the rest two types of designs, the stratified L_2-discrepancy prefers the orthogonal array-based samples than the Latin hypercube samples based on the projection properties on multi-dimensions.

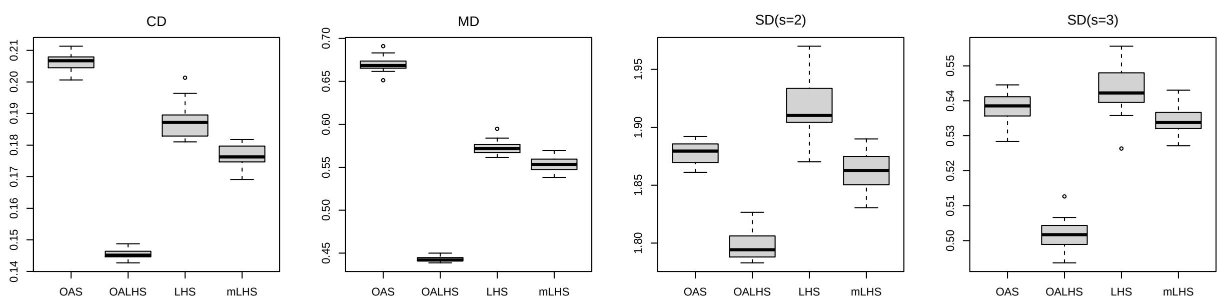

Figure 3.2: Discrepancies for four types of designs.

Example 3.5. *Consider four 25×3 Latin hypercube designs studied by Sun et al. (2019). The four designs consist of the same type of the designs as mentioned in Example 3.3. The projection plots in Sun et al. (2019) show that all four designs are quite space-filling. The uniform projection design achieves stratifications on 5×5 grids for any two-dimensional projection. We calculate the stratified L_2-discrepancy with $s = 2, 3, 5$ and other discrepancies for the four designs. All discrepancies except for the stratified L_2-discrepancy with $s = 5$ choose the uniform design as the best design. The stratified L_2-discrepancy with $s = 5$ and $p = 2$ evaluates the 5×5 stratification properties and the uniform projection design is picked out as the best design. For $s = 2$ and 3, the*

Table 3.4: Discrepancies for four 25×3 Latin hypercube designs.

Design	CD	WD	MD	SD ($s = 2$)	SD ($s = 3$)	SD ($s = 5$)
Maximin	0.0457	0.0685	0.0681	0.2197	0.1042	0.0717
MaxPro	0.0429	0.0622	0.0621	0.2027	0.1040	0.0611
Uniform	**0.0377**	**0.0620**	**0.0597**	**0.1904**	**0.0963**	0.0568
UniformPro	0.0392	0.0621	**0.0597**	0.1969	0.1027	**0.0469**

stratified L_2-discrepancy could not detect 5×5 *stratifications.*

3.4 Connection and comparisons with other criteria

3.4.1 A space-filling pattern enumerator

A minimum aberration type space-filling criterion was introduced by Tian and Xu (2021) to assess the space-filling properties of designs based on stratification properties on various grids. The space-filling criterion is based on the space-filling hierarchy principle and can systematically rank and select designs such as strong orthogonal arrays and Latin hypercubes.

For $x \in \mathbb{Z}_{s^p}$, let $\rho(x)$ be the number of digits needed to represent x in base s numeral system. We take $\rho(0) = 0$. Let $D = (d_{ij})$ be a design with n runs, m columns and all entries in $\mathbb{Z}_{s^p}$, the space-filling pattern $(S_1(D), \ldots, S_{mp}(D))$ is defined as

$$S_k(D) = n^{-2} \sum_{\substack{\rho(u)=k,\\ u\in\mathbb{Z}_{s^p}^m}} \left| \sum_{i=1}^{n} \prod_{j=1}^{m} \chi_{u_j}(d_{ij}) \right|^2, \tag{3.8}$$

where the summation is over all $u = (u_1, \cdots, u_m) \in \mathbb{Z}_{s^p}^m$ with $\rho(u) = \sum_{j=1}^m \rho(u_j) = k$ and $\chi_{u_i}(d_{ij})$ is the character defined in Tian and Xu (2021). The space-filling criterion is to select the designs that sequentially minimize the space-filling pattern.

Both the stratified L_2-discrepancy and the space-filling criterion evaluate designs based on various stratifications properties. The space-filling pattern considers the stratification properties via evaluating orthogonal properties of all possible column combinations, which results in considerable computational complexity especially in a high-dimensional space. Here we present a new way to calculate the space-filling pattern via row-wise calculations and establish a connection between these two criteria. For design $D = (d_{ij})$, define the weighted similarity between $u, v \in \mathbb{Z}_{s^p}$ as

$$R(u, v; y) = \sum_{i=0}^{s^p-1} \chi_i(u)\overline{\chi_i(v)}y^{\rho(i)},$$

where y is a parameter used to set weights and $\overline{\chi_i(v)}$ is the complex conjugate of $\chi_i(v)$. The weighted similarity between the ath row $d_a = (d_{a1}, ..., d_{am})$ and the bth row $d_b = (d_{b1}, ..., d_{bm})$ of D is

$$R(d_a, d_b; y) = \prod_{j=1}^{m} R(d_{aj}, d_{bj}; y) = \prod_{j=1}^{m} \left(\sum_{i=0}^{s^p-1} \chi_i(d_{aj})\overline{\chi_i(d_{bj})}y^{\rho(i)} \right).$$

Define the space-filling pattern enumerator for design D as

$$E(D; y) = \sum_{k=0}^{mp} S_k(D)y^k,$$

where $S_0(D) = 1$.

Lemma 3.1. *The space-filling pattern enumerator $E(D; y)$ can be computed as the average weighted similarity among all pairs of rows of D as*

$$E(D; y) = n^{-2} \sum_{a=1}^{n} \sum_{b=1}^{n} R(d_a, d_b; y) = n^{-2} \sum_{a=1}^{n} \sum_{b=1}^{n} \prod_{j=1}^{m} R(d_{aj}, d_{bj}; y).$$

The space-filling pattern enumerator is a polynomial function of y with the space-filling pattern as coefficients. The space-filling pattern can be calculated by evaluating $E(D; y)$ with different y for mp times and solving a system of linear equations. Let $y_1, \ldots, y_{mp}$ be the distinct ys chosen to compute the $E(D; y)$. A system of linear equations is

$$\begin{pmatrix} y_1 & y_1^2 & \ldots & y_1^{mp} \\ y_2 & y_2^2 & \ldots & y_2^{mp} \\ \vdots & \vdots & \ddots & \vdots \\ y_{mp} & y_{mp}^2 & \ldots & y_{mp}^{mp} \end{pmatrix} \begin{pmatrix} S_1(D) \\ S_2(D) \\ \vdots \\ S_{mp}(D) \end{pmatrix} = \begin{pmatrix} E(D; y_1) - 1 \\ E(D; y_2) - 1 \\ \vdots \\ E(D; y_{mp}) - 1 \end{pmatrix}, \tag{3.9}$$

which can be represented by $YS = E$ in a matrix form. Then the space-filling pattern can be easily solved by $S = Y^{-1}E$. Specifically, we can set $y_j = e^{2\pi j\sqrt{-1}/(mp)}$ so that $y_j^{mp} = 1$. Our choice of $y_j, j = 1, \cdots, mp$, leads to a simple inverse of the matrix $Y^{-1} = \overline{Y}^T(mp)^{-1}$ where $\overline{Y}^T$ is the complex conjugate of Y.

Theorem 3.3. *Let $y_j = e^{2\pi j\sqrt{-1}/(mp)}$. The space-filling pattern can be computed as*

$$S_k(D) = \frac{1}{mp}\sum_{j=1}^{mp} y_j^{-k}(E(D; y_j) - 1)$$

for $k = 1, \ldots, mp$.

3.4.2 Connection with the space-filling criterion

The space-filling pattern enumerator makes it possible to connect the stratified L_2-discrepancy and the space-filling criterion computationally because both the space-filling pattern enumerator and the stratified L_2-discrepancy have summation terms over all paired rows in the design. The following theorem reveals the relationship between the space-filling pattern and the stratified L_2-discrepancy.

Theorem 3.4. *For a design D with n runs, m columns and entries from $\mathbb{Z}_{s^p}$, the stratified L_2-discrepancy connects with the space-filling pattern in the following equation:*

$$SD_2^R(D)^2 = (E(D; y) - 1)(1 - y)^{-m} = \left(\sum_{i=1}^{mp} y^i S_i(D)\right)(1 - y)^{-m},$$

when the weights of the stratified L_2-discrepancy are set to $w(i) = (s^2y)^i$ if $i < p$ and $w(p) = (s^2y)^p/(1 - y)$ with $y \in (0, 1)$.

Theorem 3.4 shows that the squared stratified L_2-discrepancy is a linear combination of the space-filling pattern under the proposed weight scheme; therefore, the squared stratified L_2-discrepancy is more flexible and general than the space-filling criterion.

The proposed weight scheme in Theorem 3.4 leverages the relative importance of the local discrepancies related to the volume of region $R_u(x)$. For $u = (u_1, \ldots, u_m) \in \mathbb{Z}_{p+1}^m$, $R_u(x)$ has

Table 3.5: Space-filling pattern and the stratified L_2-discrepancy.

Design	S_4	S_5	S_6	S_7	$SD_2^R(D)$	$SD_2^R(D)(y = 0.1)$
1	27	80	248	546	2.1705	0.1074
2	27	82	242	548	2.1729	0.1076
3	28	78	250	540	2.1771	0.1084
4	28	82	238	540	2.1813	0.1087
5	29	80	240	534	2.1878	0.1097
6	29	82	234	536	2.1902	0.1098
7	30	76	248	524	2.1918	0.1105

volume $\Pi_{j=1}^m s^{-u_j}$ and weight $w(u) = \Pi_{j=1}^m w(u_j)$. When $y = s^{-2}$, $w(i) = 1$ for $1 \leq i < p$ and $w(p) = 1/(1 - s^{-2})$, which leads to the constant weight $w(u) = 1$ for all u with $\max_{j=1}^m u_j < p$. When $y < s^{-2}$, larger grids receive larger weights, which agrees with the space-filling hierarchy principle. So we recommend the choice of $y < s^{-2}$ if unequal weights are preferred.

Example 3.6. *In Table 4 of Tian and Xu (2021), the authors listed the total number of distinct space-filling patterns for all subarrays of design SOA$(32, 9, 8, 3)$ from Shi and Tang (2020). There are 7 distinct space-filling patterns from all 8-column subarrays of SOA$(32, 9, 8, 3)$. The space-filling criterion is able to rank those 7 designs with different geometric structures. We calculate the stratified L_2-discrepancy with $s = 2, p = 5$ for the 7 designs and present the results in Table 3.5, where $SD_2^R(D)$ is the stratified L_2-discrepancy with constant weights $w(u) = 1$ and $SD_2^R(D)(y = 0.1)$ is the stratified L_2-discrepancy with a weight scheme in Theorem 3.4 and $y = 0.1$. The ranking of the designs are the same for the two sets of weights. The stratified L_2-discrepancy prefers designs that achieve various stratifications when projected onto a low-dimensional space.*

Theorem 3.5. *For two designs D_1 and D_2 with n runs and m factors and entries from $\mathbb{Z}_{s^p}$, if D_1 is more space-filling than D_2 based on the space-filling criterion, there must exist an $\epsilon > 0$ such that for any $y \in (0, \epsilon)$,*

$$SD_2^R(D_1) - SD_2^R(D_2) < 0,$$

when the stratified L_2-discrepancy takes s, p and the weight scheme in Theorem 3.4.

Based on Theorem 3.5, the space-filling criterion is a special case of the stratified L_2-discrepancy. Using the weight scheme in Theorem 3.4 and a small enough y, the stratified L_2-discrepancy selects designs in the spirit of the space-filling hierarchy principle. Stratification properties of the designs are preferred from larger grids to smaller grids and stratifications on the same volume grids are treated equally important. The space-filling criterion is equivalent to the minimum G_2-aberration criterion proposed by Tang and Deng (1999) for two-level designs. The generalized minimum aberration criterion proposed by Xu and Wu (2001) for general s-level designs is also a special case of the space-filling criterion with $p = 1$. Thus, the stratified L_2-discrepancy has a deep connection with the aforementioned criteria.

Tian and Xu (2021) showed that general strong orthogonal arrays of maximum strength are space-filling under the space-filling criterion. By the connection established in Theorems 3.4 and 3.5, general strong orthogonal arrays of maximum strength tend to be uniform and have small stratified L_2-discrepancy values. This explains why the orthogonal array-based Latin hypercube samples in Example 3.4 are more uniform than other designs because they are general strong orthogonal arrays of strength two.

Despite the equivalence with the space-filling criterion as a special case, the stratified L_2-discrepancy can be applied to a much broader range of designs as long as the designs are transformed to the unit hypercube. It can serve equivalently as the space-filling criterion for design with any number of levels. The parameters s and p, as well as the weights $w(u)$, can be tuned flexibly based on the needs of the users. The stratified L_2-discrepancy provides a simple and elegant way to connect the aliasing and orthogonal properties in factorial designs with the discrepancy from the uniform designs.

3.4.3 Comparison with other discrepancies

The curse of dimensionality has been a pain for many discrepancies. The effect of dimensionality is reflected in the analytic formula. We list the expressions for the star L_2-discrepancy, centered L_2-discrepancy, wrap-around L_2-discrepancy and mixture L_2-discrepancy in the following:

$$
\begin{aligned}
D_2^*(\mathcal{P})^2 &= \left(\frac{1}{3}\right)^m - \frac{2}{n}\sum_{a=1}^{n}\prod_{j=1}^{m}\frac{(1-x_{aj}^2)}{2} + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}[1-\max(x_{aj},x_{bj})],\\
CD_2(\mathcal{P})^2 &= \left(\frac{13}{12}\right)^m - \frac{2}{n}\sum_{a=1}^{n}\prod_{j=1}^{m}\left(1+\frac{1}{2}|x_{aj}-0.5|-\frac{1}{2}|x_{aj}-0.5|^2\right)\\
&\quad + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(1+\frac{1}{2}|x_{aj}-0.5|+\frac{1}{2}|x_{bj}-0.5|-\frac{1}{2}|x_{aj}-x_{bj}|\right),\\
WD_2(\mathcal{P})^2 &= -\left(\frac{4}{3}\right)^m + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(\frac{2}{3}-|x_{aj}-x_{bj}|+|x_{aj}-x_{bj}|^2\right),\\
MD_2(\mathcal{P})^2 &= \left(\frac{19}{12}\right)^m - \frac{2}{n}\sum_{a=1}^{n}\prod_{j=1}^{m}\left(\frac{5}{3}-\frac{1}{4}|x_{aj}-0.5|-\frac{1}{4}|x_{aj}-0.5|^2\right)\\
&\quad + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(\frac{15}{8}-\frac{1}{4}|x_{aj}-0.5|-\frac{1}{4}|x_{bj}-0.5|-\frac{3}{4}|x_{aj}-x_{bj}|+\frac{1}{2}|x_{aj}-x_{bj}|^2\right).
\end{aligned}
$$

In the above expressions, some terms are related to the center point (0.5) or the end point (1). Other terms are related to the distance between two design points. In a high-dimensional space, the product of some individual terms dominates the discrepancy value, which causes the curse of dimensionality. If the terms related to special points dominate, the discrepancy will prefer points around some locations. In contrast, the expression of the stratified L_2-discrepancy in (3.7) does not include any special points; therefore, the dimension effect is not accumulated around any special point and the effect of dimensionality is eliminated.

We present an example to illustrate the dimension effect. We generate a random Latin hypercube design D_1 with 64 runs and 63 columns using R package `lhs`. Then we create two designs $D_2 = 1 - 0.25D_1$ and $D_3 = 0.25(D_1 - 0.5) + 0.5$ by relocating and scaling D_1. We further generate D_4 by $D_2 + 0.125$ and subtracting 1 from those entries larger than 1. Figure 3.3 shows the projection plots of the four designs in the first 2 dimensions. Other projection plots are similar.

Table 3.6: Logarithmic discrepancies for 4 designs.

Design	D_2^*	CD	WD	MD	SD$^*(s=2)$	SD$^*(s=3)$.
D_1	-28.4425	4.9888	**10.6903**	15.5512	**-2.4238**	**-1.6092**
D_2	**-34.6063**	9.0910	11.3280	15.4849	0.2613	0.7740
D_3	-23.8610	**2.4959**	11.3280	18.3055	-1.9591	0.7718
D_4	-34.6042	9.3556	11.3280	**14.5693**	-1.9591	-0.1387

It is obvious that the first design is more space-filling than the rest of three designs. Table 3.6 lists the logarithmic values of the discrepancies. The stratified L_2-discrepancy with star uses the weight scheme in Theorem 3.4 with $y = 0.01$.

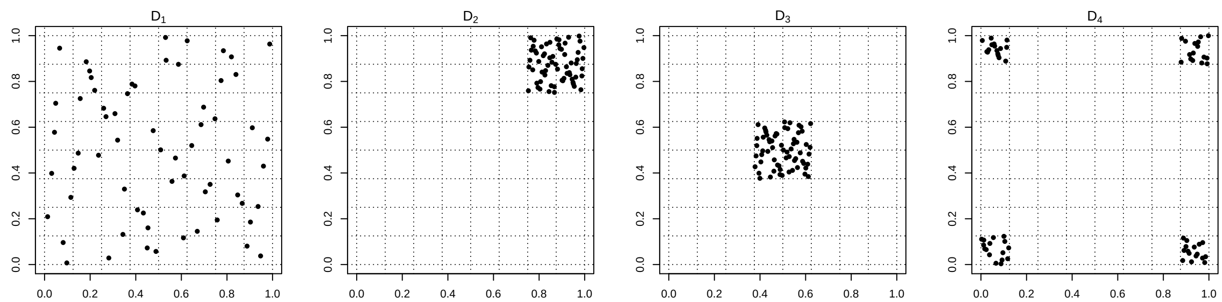

Figure 3.3: Projection plots of 4 designs in the first 2 dimensions.

The star L_2-discrepancy D_2^* chooses D_2 as the best design; the centered L_2-discrepancy chooses D_3 as the best design and the mixture L_2-discrepancy favors design D_4. When dimension gets large, these discrepancies become nonsense due to accumulated product terms. The wrap-around L_2-discrepancy and the stratified L_2-discrepancy choose D_1 as the best design. When $s = 2$, designs D_3 and D_4 have the second smallest stratified L_2-discrepancy because design points achieve basic stratification when the design region is cut to half in each dimension. When $s = 3$, D_4 has the second smallest discrepancy because not all of the design points are inside the same one third of the design region. The warp-around L_2-discrepancy has a weakness that it is invariant under location shift; therefore, it cannot distinguish D_2, D_3, D_4 even though they are quite different.

3.5 Concluding Remarks

This chapter proposes a new type of generalized L_2-discrepancy, the stratified L_2-discrepancy. The new discrepancy inherits most of the advantages of the generalized L_2-discrepancy. The curse of dimensionality faced by some of the generalized L_2-discrepancies is eliminated in our criterion. Projection properties of the designs are assessed by evaluating stratification properties on various grids with flexible weights. Users can define how the design region is stratified and what principle to follow when selecting space-filling designs. The space-filling criterion is connected with the stratified L_2-discrepancy via the space-filling pattern enumerator. As a result, the stratified L_2-discrepancy extends the usage of the space-filling criterion to a broader range of designs as long as the designs are transformed to the unit hypercube. The stratified L_2-discrepancy provides a more general and flexible way to evaluate space-filling designs based on stratification.

The stratification of the design region opens a new topic in assessing space-filling properties. Connections can be seen between digital nets, (t, s)-sequence and the stratified L_2-discrepancy. The asymptotic property of the stratified L_2-discrepancy is an interesting topic to discover. So far, we list one possible weight scheme depending on the volume of the grids. New weight awaits exploring to consider both the size of the grids and the number of dimensions due to the fact that there is often only limited number of active factors in most experiments. The rules of thumb for choosing s and p could be further investigated by conducting more experiments to firm up a robust suggestion for their choices. The construction of uniform designs remains challenging. A possible approach is to search for designs with good stratified L_2-discrepancy in an algorithmic way. However, the update formula needs to be derived to accelerate the existing algorithms.

3.6 Appendix: Proofs

Proof of Theorem 3.1. Let F be the cumulative distribution function for the uniform distribution and let $F_{\mathcal{P}}$ be the empirical distribution function for design $\mathcal{P}$, the local projection discrepancy

(3.1) can be expanded as

$$\begin{aligned}
\text{disc}_u^R(x) &= \text{Vol}(R_u(x)) - \frac{|\mathcal{P} \cap R_u(x)|}{n} \\
&= \int_{C^m} 1_{R_u(x)}(t)dF(t) - \int_{C^m} 1_{R_u(x)}(t)dF_{\mathcal{P}}(t) \\
&= \int_{C^m} 1_{R_u(x)}(t)d(F - F_{\mathcal{P}})(t).
\end{aligned}$$

Integrating the squared local projection discrepancy over the design region with weight $w(u)$, we have

$$\begin{aligned}
\int_{C^m} w(u)|\text{disc}_u^R(x)|^2 dx &= \int_{C^m} w(u) \int_{C^{2m}} 1_{R_u(x)}(t)1_{R_u(x)}(z)d(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z)dx \\
&= \int_{C^{2m}} \int_{C^m} w(u)1_{R_u(x)}(t)1_{R_u(x)}(z)dxd(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z) \\
&= \int_{C^{2m}} \mathcal{K}_u^R(t, z)d(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z).
\end{aligned}$$

Then the stratified L_2-discrepancy (3.3) can be written as follows:

$$\begin{aligned}
SD_2^R(\mathcal{P}) &= \left\{ \int_{C^m} \sum_{u \in \mathbb{Z}_{p+1}^m} w(u)|\text{disc}_u^R(x)|^2 dx \right\}^{\frac{1}{2}} \\
&= \left\{ \sum_{u \in \mathbb{Z}_{p+1}^m} \int_{C^{2m}} \mathcal{K}_u^R(t, z)d(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z) \right\}^{\frac{1}{2}} \\
&= \left\{ \int_{C^{2m}} \sum_{u \in \mathbb{Z}_{p+1}^m} \mathcal{K}_u^R(t, z)d(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z) \right\}^{\frac{1}{2}} \\
&= \left\{ \int_{C^{2m}} \mathcal{K}^R(t, z)d(F - F_{\mathcal{P}})(t)d(F - F_{\mathcal{P}})(z) \right\}^{\frac{1}{2}} \\
&= \|F - F_{\mathcal{P}}\|_{\mathcal{W}}.
\end{aligned}$$

Theorem 3.1 is proved. The stratified L_2-discrepancy can be treated as a member of the generalized L_2-discrepancy family with modified summation of local projection discrepancy and weights. $\square$

Proof of Theorem 3.2. The analytic expression of the stratified L_2-discrepancy is obtained through integration over the simplified kernel functions for each dimension. The kernel function is

$$\mathcal{K}^R(t,z) = \prod_{j=1}^{m}[1+\sum_{i=1}^{p}\tilde{\mathcal{K}}_i^R(t_j,z_j)] = \prod_{j=1}^{m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^i}\delta_i(t_j,z_j)].$$

The integration of $\mathcal{K}^R(t,z)$ over $C^m \times C^m$ for the uniform distribution is

$$\begin{aligned}\int_{C^{2m}}\mathcal{K}^R(t,z)dF(t)dF(z) &= \int_{C^{2m}}\mathcal{K}^R(t,z)dtdz \\ &= \int_{C^{2m}}\prod_{j=1}^{m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^i}\delta_i(t_j,z_j)]dtdz \\ &= \int_{C^m}\prod_{j=1}^{m}\left(\int_{C^m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^i}\delta_i(t_j,z_j)]dt\right)dz.\end{aligned}$$

By the definition of $\delta_i(t_j,z_j)$ in (3.6), we have $\int_{C^m}\delta_i(t_j,z_j)dt = s^{-i}$ and so

$$\begin{aligned}\int_{C^{2m}}\mathcal{K}^R(t,z)dF(t)dF(z) &= \int_{C^m}\prod_{j=1}^{m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^{2i}}]dz \\ &= \prod_{j=1}^{m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^{2i}}].\end{aligned}$$

For a fixed row $x_a \in C^m$ of design $\mathcal{P}$, the integration of $\mathcal{K}^R(t,x_a)$ over C^m is

$$\int_{C^m}\mathcal{K}^R(t,x_a)dF(t) = \int_{C^m}\mathcal{K}^R(t,x_a)dt = \prod_{j=1}^{m}[1+\sum_{i=1}^{p}\frac{w(i)}{s^{2i}}].$$

The expansion of the integral of $SD_2^R(\mathcal{P})^2$ consists of three terms we computed above.

$$\begin{aligned}SD_2^R(\mathcal{P})^2 &= \int_{C^{2m}}\mathcal{K}^R(t,z)d(F-F_{\mathcal{P}})(t)d(F-F_{\mathcal{P}})(z) \\ &= \int_{C^{2m}}\mathcal{K}^R(t,z)dF(t)dF(z) - \frac{2}{n}\sum_{a=1}^{n}\int_{C^m}\mathcal{K}^R(t,x_a)dF(t) + \frac{1}{n^2}\sum_{a,b=1}^{n}\mathcal{K}^R(x_a,x_b)\end{aligned}$$

Thus, the analytical expression of the stratified L_2-discrepancy is straightforward. □

Proof of Lemma 3.1. Based on the definition of $\rho(i), i \in \mathbb{Z}_{s^p}$, the weighted similarity between the ath row $d_a = (d_{a1},...,d_{am})$ and the bth row $d_b = (d_{b1},...,d_{bm})$ of design D is a polynomial function of y of order pm. Thus, it follows that

$$\prod_{j=1}^{m} R(d_{aj}, d_{bj}) = \prod_{j=1}^{m} \left(\sum_{i=0}^{s^p-1} \chi_i(d_{aj}) \overline{\chi_i(d_{bj})} y^{\rho(i)} \right)$$
$$= \sum_{k=0}^{mp} \left(\sum_{\rho(i_1)+...+\rho(i_j)=k} \prod_{j=1}^{m} \chi_{i_j}(d_{aj}) \overline{\chi_{i_j}(d_{bj})} \right) y^k,$$

where the second summation is over all integer combinations $(i_1, ..., i_j) \in \mathbb{Z}_{s^p}^m$ that achieve $\rho(i_1) + ... + \rho(i_j) = k$. The summation of the weighted similarity over all pairs of rows in D can be derived as

$$n^{-2} \sum_{a=1}^{n} \sum_{b=1}^{n} \prod_{j=1}^{m} R(d_{aj}, d_{bj}) = n^{-2} \sum_{a=1}^{n} \sum_{b=1}^{n} \sum_{k=0}^{mp} \left(\sum_{\rho(i_1)+...+\rho(i_j)=k} \prod_{j=1}^{m} \chi_{i_j}(d_{aj}) \overline{\chi_{i_j}(d_{bj})} \right) y^k$$
$$= n^{-2} \sum_{k=0}^{mp} \left(\sum_{\rho(i_1)+...+\rho(i_j)=k} \sum_{a=1}^{n} \sum_{b=1}^{n} \prod_{j=1}^{m} \chi_{i_j}(d_{aj}) \overline{\chi_{i_j}(d_{bj})} \right) y^k$$
$$= n^{-2} \sum_{k=0}^{mp} \sum_{\rho(i_1)+...+\rho(i_j)=k} \left| \sum_{a=1}^{n} \prod_{j=1}^{m} \chi_{i_j}(d_{aj}) \right|^2 y^k$$
$$= \sum_{k=0}^{mp} S_k(D) y^k.$$

Thus, the space-filling pattern enumerator can be computed as the average weighted similarity among all pairs of rows. □

Proof of Theorem 3.4. By Theorem 3.2, there are two terms in the analytic expression of $SD_2^R(\mathcal{P})^2$. Using the weight scheme stated in Theorem 3.4, the first term multiplied by $(1-y)^m$ is

$$-\left(1 + \sum_{i=1}^{p} w(i) s^{-2i}\right)^m (1-y)^m = -\left(1 - y + (1-y) \sum_{i=1}^{p-1} y^i + y^p\right)^m = -1.$$

The second term multiplied by $(1-y)^m$ is

$$\begin{aligned}
&\frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(1+\sum_{i=1}^{p}\frac{w(i)}{s^i}\delta_i(x_{aj},x_{bj})\right)(1-y)^m \\
&=\frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(1-y+\sum_{i=1}^{p-1}s^iy^i(1-y)\delta_i(x_{aj},x_{bj})+s^py^p\delta_p(x_{aj},x_{bj})\right) \\
&=\frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(1+\sum_{i=1}^{p}y^is^{i-1}\delta_{i-1}(x_{aj},x_{bj})(s\delta_i(x_{aj},x_{bj})-1)\right). \qquad (3.10)
\end{aligned}$$

Based on Lemma 3.1, the space-filling pattern enumerator can be computed as

$$E(D;y)=\frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}R(x_{aj},x_{bj})=\frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\left(\sum_{i=0}^{s^p-1}\chi_i(x_{aj})\overline{\chi_i(x_{bj})}y^{\rho(i)}\right). \qquad (3.11)$$

In order to prove

$$SD_2^R(D)^2(1-y)^m=E(D;y)-1,$$

we only need to prove (3.10) = (3.11). Both equations have double summations and products over all pairs of rows and all columns. Thus, we need to prove

$$1+\sum_{i=1}^{p}y^is^{i-1}\delta_{i-1}(x_{aj},x_{bj})(s\delta_i(x_{aj},x_{bj})-1)=\sum_{i=0}^{s^p-1}\chi_i(x_{aj})\overline{\chi_i(x_{bj})}y^{\rho(i)}. \qquad (3.12)$$

According to Tian and Xu (2021), the character $\chi_u(x)$ is orthonormal and symmetric such that

$$\begin{aligned}
\sum_{x\in\mathbb{Z}_{s^p}}\chi_u(x)\overline{\chi_v(x)} &= \sum_{x\in\mathbb{Z}_{s^p}}\chi_x(u)\overline{\chi_x(v)} \\
&=\sum_{x\in\mathbb{Z}_{s^p}}\prod_{i=1}^{p}\xi^{(f_i(u)-f_i(v))f_{p+1-i}(x)} \\
&=\prod_{i=1}^{p}\sum_{f_{p+1-i}(x)\in\mathbb{Z}_s}\xi^{(f_i(u)-f_i(v))f_{p+1-i}(x)} \qquad (3.13) \\
&=\begin{cases}|\mathbb{Z}_{s^p}| & u=v \\ 0 & u\neq v\end{cases}, u,v\in\mathbb{Z}_{s^p},
\end{aligned}$$

where $\xi=e^{2\pi\sqrt{-1}/s}$ and $f_i(x)=\lfloor x/s^{p-i}\rfloor \pmod s$ as defined in Tian and Xu (2021) for $i=1,\ldots,p$ and $x\in\mathbb{Z}_{s^p}$. Equation (3.13) holds because when x loops through $\mathbb{Z}_{s^p}$, $f_i(x), i=1,\cdots,p$

also loops through each value in $\mathbb{Z}_s$. We can express the right side of Equation (3.12) as $\sum_{k=0}^{p} c_k y^k$, where the coefficient c_k is

$$\sum_{\substack{\rho(u)=k\\ u\in\mathbb{Z}_{s^p}}} \chi_u(x_{aj})\overline{\chi_u(x_{bj})} = \sum_{\substack{\rho(u)=k\\ u\in\mathbb{Z}_{s^p}}} \prod_{i=1}^{p} \xi^{(f_i(x_{aj})-f_i(x_{bj}))f_{p+1-i}(u)}$$

$$= \sum_{\substack{\rho(u)=k\\ u\in\mathbb{Z}_{s^p}}} \prod_{i=1}^{k} \xi^{(f_i(x_{aj})-f_i(x_{bj}))f_{p+1-i}(u)}. \tag{3.14}$$

Equation (3.14) is based on the definition of $\rho(u) = p + 1 - \min\{i|f_i(u) \neq 0, i = 1, \dots, p\}$ for $u \in \mathbb{Z}_{s^p}$. The set of u with $\rho(u) = k$ is

$$\begin{aligned}\{u \in \mathbb{Z}_{s^p}|f_i(u) = 0, i \leq p - k;\\ f_i(u) = \{1, \cdots, s-1\}, i = p+1-k;\\ f_i(u) = \{0, \cdots, s-1\}, i > p+1-k\}.\end{aligned}$$

Equation (3.14) can be written as a product:

$$(3.14) = \left(\prod_{i=1}^{k-1} \sum_{f_{p+1-i}(u)\in\mathbb{Z}_s} \xi^{(f_i(x_{aj})-f_i(x_{bj}))f_{p+1-i}(u)}\right) \tag{3.15}$$

$$\times \left(\sum_{f_{p+1-k}(u)=1}^{s-1} \xi^{(f_k(x_{aj})-f_k(x_{bj}))f_{p+1-k}(u)}\right) \tag{3.16}$$

$$= s^{k-1}\delta_{k-1}(x_{aj}, x_{bj})(s\delta_k(x_{aj}, x_{bj}) - 1). \tag{3.17}$$

In Equation (3.15, 3.16), we split the summation of the product in Equation (3.14) to product of summation in the similar way we did in Equation (3.13). Comparing (3.15) with (3.13), we get the term in (3.15) equal to $s^{k-1}\delta_{k-1}(x_{aj}, x_{bj})$. The term in Equation (3.16) is a summation over $f_{p+1-k}(u) = 1, \cdots, s-1$ which takes values $s-1$ when $f_k(x_{aj}) - f_k(x_{bj}) = 0$. When $f_k(x_{aj}) \neq f_k(x_{bj})$, we know that

$$\sum_{f_{p+1-k}(u)=0}^{s-1} \xi^{(f_k(x_{aj})-f_k(x_{bj}))f_{p+1-k}(u)} = \sum_{f_{p+1-k}(u)=1}^{s-1} \xi^{(f_k(x_{aj})-f_k(x_{bj}))f_{p+1-k}(u)} + 1 = 0$$

because the character is orthonormal when $p = 1$. Thus, the term in Equation (3.16) is $s\delta_k(x_{aj}, x_{bj}) - 1$. Finally, Equation (3.17) is the coefficient before y^k on the left side of Equation (3.12). The proof is complete.

$\square$

CHAPTER 4

Construction of Optimal Space-Filling Designs based on Stratification

The space-filling criterion and the stratified L_2-discrepancy have been proposed to measure the space-filling properties of designs based on stratification properties. Both criteria show effectiveness in selecting space-filling designs. However, the construction of optimal designs is not yet discussed. In this chapter, we develop a lower bound for the stratified L_2-discrepancy and the space-filling pattern enumerator. We define a metric space to characterize the distance between points in a stratification view so that the stratified L_2-discrepancy becomes a criterion based on distance. The space-filling pattern enumerator is a special case of the stratified L_2-discrepancy, thus we obtain another lower bound. We consider the construction of optimal designs based on the conditions of which the lower bounds hold. The optimal designs are space-filling under other criteria and are constructed a lot faster than algorithmic search especially in high-dimensional spaces. Examples of the optimal designs are given.

4.1 Notation and Backgrounds

The stratification properties of a general strong orthogonal array D are characterized in the space-filling criterion proposed by Tian and Xu (2021). The space-filling pattern $(S_1(D), \ldots, S_{mp}(D))$ is defined to be a vector, each element of which reveals how space-filling the design points are on stratifications with certain volume grids. We rank and select designs that sequentially minimize the elements of the space-filling pattern. For example, a general strong orthogonal array of strength t

has $S_i(D) = 0, i = 1, \cdots, t$. General strong orthogonal arrays of maximum strength are preferred by the space-filling criterion.

The space-filling criterion is restricted to designs with a fixed number of levels. Attempts to gather stratification information have been made to uniform designs as well. The stratified L_2-discrepancy has been proposed to evaluate uniform designs based on the stratification properties flexibly. The discrepancy is general and powerful in the following aspects: (i) it is applicable to all designs as long as their entries are transformed to $C = [0, 1]$; (ii) stratification scheme and the importance level assigned to each type of stratification could be tuned freely. The stratified L_2-discrepancy is a generalization of the space-filling pattern. The space-filling criterion follows the space-filling hierarchy principle, while the stratified L_2-discrepancy could have different preferences by assigning weights to each type of stratifications.

Connection between the stratified L_2-discrepancy and the space-filling pattern is established via the space-filling pattern enumerator $E(D; y)$, which is a polynomial with coefficients being the space-filling pattern as follows:

$$E(D; y) = \sum_{k=0}^{mp} S_k(D) y^k. \tag{4.1}$$

The space-filling pattern enumerator $E(D; y)$ has a fast computational formula so it makes the computation of the space-filling pattern much easier. When y is small enough, using the space-filling pattern to rank designs is equivalent to using the value of $E(D; y)$. $E(D; y)$ and the stratified L_2-discrepancy with special weights uniquely determine each other. We show a lower bound for the stratified L_2-discrepancy is helpful in finding optimal space-filling designs based on stratification properties.

4.2 A Lower Bound for the Stratified L_2-discrepancy

4.2.1 The NRT-distance

General strong orthogonal arrays are closely related to (t, m, s)-nets proposed by Niederreiter (1987a) for quasi-Monte Carlo methods of numerical integration based on work from Schmid (1995); Mullen and Schmid (1996) and Martin and Stinson (1999). The common factorial designs have fruitful theoretical results based on coding theory and Hamming space. Yet no stratification information is obtained. Bierbrauer et al. (2002) proposed the NRT-space to generalize the Hamming space. The NRT-space is named after the work in Niederreiter (1986, 1987b) and Rosenbloom and Tsfasman (1997).

We generalize the setting of the NRT-space in this chapter in design language. Given positive integers s and p, we first define a series of helper functions $f_i(x), i = 1, \cdots, p$:

$$f_i(x) = \begin{cases} \lfloor x/s^{p-i} \rfloor \mod s & x \in \mathbb{Z}_{s^p} \\ \lfloor \lfloor s^p x \rfloor / s^{p-i} \rfloor \mod s & x \in [0,1) \end{cases}.$$

Suppose the stratification parameters are s and p, the series of functions give out p digits of x in base s numeral system. The base s representation of x is expressed as $(f_1 f_2 \cdots f_p)_s$ if $x \in \mathbb{Z}_{s^p}$ or $(0.f_1 f_2 \cdots f_p \cdots)_s$ if $x \in [0,1)$. To transform back, $x = \sum_{k=1}^{p} f_k(x) s^{p-k}$ if $x \in \mathbb{Z}_{s^p}$ or $x = \sum_{k=1}^{\infty} f_k(x) s^{-k}$ if $x \in [0,1)$. For instance, suppose $s = 2, p = 3$, the base 2 representation of $x = 2$ is $(010)_2$ with $f_1(2) = f_3(2) = 0, f_2(2) = 1$. The base 2 representation of $x = 3/8$ is $(0.011)_2$ with $f_1(3/8) = 0, f_2(3/8) = f_3(3/8) = 1$. Although the definition of the $f_i(x)$ is different for $x \in \mathbb{Z}_{s^p}$ or $x \in [0,1)$, the information they provide is the same. They both reveal the location of x when the design region is stratified to $s^0, s^1, s^2, \cdots, s^p$ parts.

Definition 4.1. *A metric space over $\mathbb{Z}_{s^p}$ or $[0,1)$ is called an NRT-space if it is endowed with the NRT-distance $\rho(x, y)$ defined by*

$$\rho(x, y) = p + 1 - \min\{i | f_i(x) - f_i(y) \neq 0, i = 1, \ldots, p\} \quad x \neq y$$

and $\rho(x, x) = 0$.

It is obvious that the NRT-distance is non-negative and symmetric. The triangle inequality holds $\rho(x,y) \le \rho(x,z) + \rho(y,z)$. We can prove this by the fact that $\min\{i|f_i(x) - f_i(y) \neq 0\} \ge \min\left(\min\{i|f_i(x) - f_i(z) \neq 0\}, \min\{i|f_i(y) - f_i(z) \neq 0\}\right)$ when $x \neq y \neq z$. If $x = y$ or $x = z$, the triangle inequality holds. For stratification purpose, the identity of indiscernibles is valid.

The NRT-distance reveals the level of closeness between two points based on stratification. An NRT-distance of d indicates that the two points will fall into the same grid when the design region is stratified to s^{p-d} parts and will not stay in the same grid for any stratification thinner than it among stratifying the design region to $s^0, s^1, s^2, \cdots, s^p$ parts. The NRT-distance provides a measure to numerically characterize the distance based on the stratifications. There are $p+1$ possible NRT-distances. We present some examples to illustrate the idea.

Example 4.1. *Suppose that the NRT-space is over $\mathbb{Z}_{2^3}$, $x = 4$, $y = 6$ and the NRT-distance between x, y is*

$$\rho(x,y) = 4 - \min\{i|f_i(x) - f_i(y) \neq 0, i = 1, \ldots, 3\}.$$

The base 2 representation of x, y are $(100)_2$ and $(110)_2$ respectively. The first i such that $f_i(x) - f_i(y) \neq 0$ is 2. Thus, the NRT-distance $\rho(4,6) = 4 - 2 = 2$. The design region $\mathbb{Z}_{2^3}$ can be stratified to 1, 2, 4 and 8 parts. The numbers $x = 4$ and $y = 6$ are in the same grid when $\mathbb{Z}_{2^3}$ is divided to 2 parts but will not when $\mathbb{Z}_{2^3}$ is divided to 4 parts. The left plot in Figure 4.1 shows the NRT-distance between $x, y \in \mathbb{Z}_{2^3}$ on horizontal and vertical axes.

Example 4.2. *Suppose that the NRT-space is over $[0,1)$ with stratification parameters $s = 3, p = 2$ so that there are 3 possible NRT-distances: 0, 1, 2. For any $x \in [0, 1/9)$, the NRT-distance $\rho(x,y) = 0$ if $y \in [0, 1/9)$ as both points are in the same one ninth grid. If $y \in [1/9, 1/3)$, the NRT-distance $\rho(x,y) = 1$ because the two points are in the same one third grid but not in the same one ninth grid. If $y \in [1/3, 1)$, the NRT-distance $\rho(x,y) = 2$. The right plot in Figure 4.1 shows the NRT-distance between $x, y \in [0,1)$ on horizontal and vertical axes. Each coordinate is stratified to 9 grids. It is interesting that there are more grids but fewer possible distances compared with the plot on the left. The choice of s and p impacts the distance and analyses.*

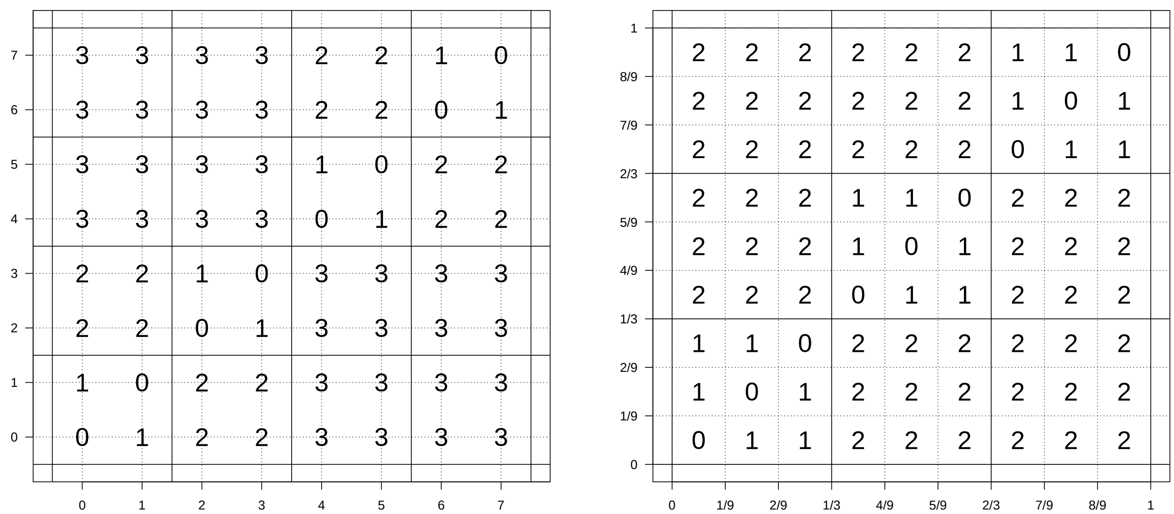

Figure 4.1: NRT-distance for $s = 2, p = 3$ (left) and $s = 3, p = 2$ (right).

4.2.2 A Lower Bound for the Stratified L_2-discrepancy

For design $\mathcal{P} = (x_{ij})_{n\times m}$ with entries from $C = [0, 1)^m$, or a q-level design $D = (d_{ij})$ with entries from $\mathbb{Z}_q$ transformed to $\mathcal{P} = (x_{ij})$ with $x_{ij} = (d_{ij} + 0.5)/q$, the stratified L_2-discrepancy with parameter s, p and weights $w(i), i = 1, \cdots, p$, has the following expression:

$$\begin{aligned} SD_2^R(\mathcal{P})^2 &= -\left(1 + \sum_{i=1}^{p} w(i)s^{-2i}\right)^m + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\sum_{i=0}^{p} w(i)s^{-i}\delta_i(x_{aj}, x_{bj}) \\ &= -\left(1 + \sum_{i=1}^{p} w(i)s^{-2i}\right)^m + \frac{1}{n^2}\sum_{a,b=1}^{n}\prod_{j=1}^{m}\mathcal{K}_j(\mathcal{P}; a, b), \end{aligned} \tag{4.2}$$

where

$$\delta_i(x_{aj}, x_{bj}) = \prod_{k=1}^{i} 1_{f_k(x_{aj})=f_k(x_{bj})}$$

for $i = 1, \cdots, p$ are the indicator functions revealing whether x_{aj} and x_{bj} are in the same grid when $[0, 1)$ is stratified to s^i parts. We define $w(0) = 1$ and $\delta_0(x_{aj}, x_{bj}) = 1$ to make the formula consistent. If $\delta_i(x, y) = 1$, we have $\delta_k(x, y) = 1$ for $1 \le k \le i$. We illustrate the connection between the values of $\delta_i(x, y)$ and $\rho(x, y)$ in the following lemma.

Lemma 4.1. *The NRT-distance $\rho(x,y)$ and indicator function $\delta_i(x,y)$ uniquely determine each other as follows:*

$$\rho(x,y) = p - \max_i\{\delta_i(x,y)=1\}.$$

The NRT-distance between x, y and values of $\delta_i(x,y)$ both present the relative position of x, y based on stratification. They are equivalent by simple algebraic derivation.

The $\mathcal{K}_j(\mathcal{P}; a, b)$ in the second part of Equation (4.2) is determined by the values of $\delta_i(x_{aj}, x_{bj}), i = 1, \cdots, p$, and so it is determined by the NRT-distance between x_{aj}, x_{bj}. For design $\mathcal{P}$, denote

$$\begin{aligned} n_k(a,b) &= \#\{l : \rho(x_{al}, x_{bl}) = k, l = 1, \cdots, m\}, \quad 1 \le a \ne b \le n, k = 0, \cdots, p \\ \sigma_k &= \mathcal{K}_j(\mathcal{P}; a, b) \quad \text{if} \;\; \rho(x_{aj}, x_{bj}) = k \\ &= \sum_{i=0}^{p-k} w(i) s^{-i}, \quad k = 0, \cdots, p. \end{aligned}$$

The stratified L_2-discrepancy in (4.2) can be represented by $n_k(a,b)$ and σ_k as

$$SD_2^R(\mathcal{P})^2 = -\left(1 + \sum_{i=1}^{p} w(i) s^{-2i}\right)^m + \frac{1}{n}\sigma_0^m + \frac{1}{n^2}\sum_{a=1}^{n}\sum_{b\ne a}^{n}\prod_{k=0}^{p}\sigma_k^{n_k(a,b)}. \tag{4.3}$$

Theorem 4.1. *For design $\mathcal{P} = (x_{ij})_{n\times m}$ with entries from C, if the elements of each column of $\mathcal{P}$ spread evenly when C is stratified to s^p grids, the stratified L_2-discrepancy $SD_2^R(\mathcal{P})$ with parameter s, p satisfies*

$$SD_2^R(\mathcal{P})^2 \ge -\left(1 + \sum_{i=1}^{p} w(i) s^{-2i}\right)^m + \frac{1}{n}\left[\sigma_0^m + (n-1)\left(\sigma_0^{\frac{nm}{s^p(n-1)} - \frac{m}{n-1}} \prod_{k=1}^{p} \sigma_k^{\frac{s^{k-1}(s-1)nm}{s^p(n-1)}}\right)\right].$$

The equality holds if and only if $n_0(a,b) = \frac{nm}{s^p(n-1)} - \frac{m}{n-1}$ and $n_k(a,b) = \frac{s^{k-1}(s-1)nm}{s^p(n-1)}$ for $k = 1, \cdots, p$ and any pair of a and b.

The lower bound of the stratified L_2-discrepancy with parameter s, p is developed for designs each column of which has entries uniformly distributed over s^p equally divided grids on C. We call these designs balanced designs analogue to balanced designs with a fixed number of levels. Balanced designs achieve uniformity in any one-dimensional projection. Latin hypercube designs and orthogonal arrays with strength as least 1 are balanced designs. It is natural to move the notion to designs on the unit hypercube.

The lower bound of the stratified L_2-discrepancy is achieved when $n_0(a,b), \cdots, n_p(a,b)$ take certain values for any pair of rows a, b. We have $n_0(a,b) + \cdots + n_p(a,b) = m$. Suppose $n = \lambda s^p$, the conditions to achieve the lower bound of the stratified L_2-discrepancy are

$$n_0(a,b) = \frac{(\lambda - 1)m}{(n-1)}, \quad n_k(a,b) = \frac{\lambda s^{k-1}(s-1)m}{(n-1)}, k = 1, \cdots, p. \tag{4.4}$$

The conditions are not related to the weights of the stratified L_2-discrepancy. In other words, the designs that achieve the lower bound of the stratified L_2-discrepancy are optimal regardless of the weights.

4.2.3 A Lower Bound for the Space-Filling Pattern Enumerator for Design with s^p Levels

The space-filling pattern enumerator $E(D; y)$ is proposed to connect the space-filling criterion (Tian and Xu, 2021) and the stratified L_2-discrepancy. The stratified L_2-discrepancy $SD_2^R(\mathcal{P})$ with special weights and $E(D; y)$ uniquely determine each other shown in the following lemma.

Lemma 4.2. *For a design D with n runs, m columns and entries from $\mathbb{Z}_{s^p}$, the stratified L_2-discrepancy connects with the space-filling pattern in the following equation:*

$$SD_2^R(D)^2 = (E(D; y) - 1)(1 - y)^{-m} = \left(\sum_{i=1}^{mp} y^i S_i(D) \right) (1 - y)^{-m},$$

when the weights of the stratified L_2-discrepancy are set to $w(i) = (s^2 y)^i$ if $i < p$ and $w(p) = (s^2 y)^p / (1 - y)$ with $y \in (0, 1)$.

The space-filling pattern enumerator forms a bridge between the space-filling criterion and the stratified L_2-discrepancy. On one hand, the space-filling pattern enumerator defined in Equation (4.1) could be used to rank designs equivalently to the space-filling criterion with a small enough y. On the other hand, the space-filling pattern enumerator is monotonically related to the stratified L_2-discrepancy with special weights. A lower bound of the space-filling pattern enumerator can be derived based on Theorem 4.1 and Lemma 4.2. The definition of $n_k(a,b)$ holds for design D using the NRT-distance defined over $\mathbb{Z}_{s^p}$. However, we need to redefine σ_k with the weights in Lemma 4.2 and a multiplier $1-y$.

$$\sigma_k = \begin{cases} \sum_{i=0}^{p-1} s^i y^i (1-y) + s^p y^p & k = 0 \\ \sum_{i=0}^{p-k} s^i y^i (1-y) & k = 1, \cdots, p \end{cases}.$$

Theorem 4.2. *For a balanced design D with n runs, m factors and entries from $\mathbb{Z}_{s^p}$, the space-filling pattern enumerator $E(D;y)$ satisfies*

$$E(D;y) \geq \frac{1}{n}\left[\sigma_0^m + (n-1)\left(\sigma_0^{\frac{nm}{s^p(n-1)} - \frac{m}{n-1}} \prod_{k=1}^{p} \sigma_k^{\frac{s^{k-1}(s-1)nm}{s^p(n-1)}}\right)\right]. \tag{4.5}$$

The equality holds if and only if $n_0(a,b) = \frac{nm}{s^p(n-1)} - \frac{m}{n-1}$ and $n_k(a,b) = \frac{s^{k-1}(s-1)nm}{s^p(n-1)}$ for $k = 1, \cdots, p$ and any pair of a and b.

The optimal designs for the stratified L_2-discrepancy are optimal designs under the space-filling criterion. We list some concrete examples on the values of σs.

Example 4.3. *When $s = 2$ and $p = 2$, there are 3 possible NRT-distances: $\rho(x,y) \in \{0,1,2\}, x, y \in \mathbb{Z}_{2^2}$. The values of σs are*

$$\sigma_0 = 1 + y + 2y^2, \quad \sigma_1 = 1 + y - 2y^2, \quad \sigma_2 = 1 - y.$$

When $s = 2$ and $p = 3$, there are 4 possible NRT-distances: $\rho(x,y) \in \{0,1,2,3\}$ for $x, y \in \mathbb{Z}_{2^3}$. The values of σs are

$$\sigma_0 = 1 + y + 2y^2 + 4y^3, \quad \sigma_1 = 1 + y + 2y^2 - 4y^3, \quad \sigma_2 = 1 + y - 2y^2, \quad \sigma_3 = 1 - y.$$

When $s = 3$ and $p = 2$, there are 3 possible NRT-distances: $\rho(x,y) \in \{0,1,2\}, x, y \in \mathbb{Z}_{3^2}$. The values of σs are

$$\sigma_0 = 1 + 2y + 6y^2, \quad \sigma_1 = 1 + 2y - 3y^2, \quad \sigma_2 = 1 - y.$$

The lower bound of the space-filling pattern enumerator provides hints on the lower bound of $S_2(D)$ in the space-filling pattern as $S_1(D) = 0$ for balanced designs. $E(D;y)$ is a polynomial function of y with space-filling pattern as coefficients. The lower bound of $E(D;y)$ is also a polynomial function of y. Comparing the coefficients, we present the following result:

Theorem 4.3. *For a balanced design D with n runs, m factors and entries from $\mathbb{Z}_{s^p}$. When $s = 2$, we can simplify the lower bound of $E(D;y)$ to*

$$E(D;y) \geq 1 + \frac{m(m+1-n)}{2(n-1)}y^2 + \cdots + \frac{(s^{p-1}(s-1))^m}{n}y^{pm}.$$

When $s = 3$, similarly, we have

$$E(D;y) \geq 1 + \frac{m(2m+1-n)}{(n-1)}y^2 + \cdots + \frac{(s^{p-1}(s-1))^m}{n}y^{pm}.$$

As y goes to 0, Theorem 4.3 provides lower bounds for $S_2(D)$ to be $\frac{m(m+1-n)}{2(n-1)}$ for designs with entries from $\mathbb{Z}_{2^p}$ and $\frac{m(2m+1-n)}{(n-1)}$ for designs with entries from $\mathbb{Z}_{3^p}$. Optimal designs that achieve the lower bound in Theorem 4.2 have $S_2(D)$ that reach above lower bounds.

4.3 Construction of Optimal Designs

4.3.1 Generalized Hadamard Matrices

We consider finding optimal designs with a fixed number of levels. Specially, we consider designs with entries from Galois field $GF(s^p)$ with operations of multiplication, addition, subtraction and division properly defined where s is a prime. The NRT-distance is valid on $GF(s^p)$. The subtraction operation is implicitly used in the definition of the NRT-distance.

Lemma 4.3. *For any $x, y \in GF(s^p)$, the NRT-distance between x and y is the same as the NRT-distance between $x - y$ and 0.*

$$\rho(x,y) = \rho(x-y,0).$$

We call $\rho(x,0) = \rho(x)$ as the weight of x over $GF(s^p)$

Elements of $GF(s^p)$ can be represented by polynomials of x with degree strictly less than p. For $y \in GF(s^p)$, the polynomial representation of y is $\sum_{k=1}^{p} f_i(y)x^{k-1}$. Addition and subtraction on $GF(s^p)$ are the usual addition and subtraction on the polynomials modulo s. Here is a simple example. Suppose $y_1 = 4$ and $y_2 = 6$ over $GF(2^3)$, the base-2 representation of y_1 and y_2 are $(100)_2$ and $(110)_2$ respectively. The polynomial representations of y_1 and y_2 are $y_1 = x^2$ and $y_2 = x^2 + x$; so $y_2 - y_1 = x$, which is $(010)_2$.

Lemma 4.4. *The weight distribution of elements in $GF(s^p)$ is*

$$n_k = \#\{x \in GF(s^p)|\rho(x) = k\} = s^{k-1}(s-1), \quad k = 1, \cdots, p$$

and $n_0 = 1$.

Combining the result in Lemmas 4.3 and 4.4, we present one class of optimal designs that achieve the lower bound of $E(D; y)$ in Theorem 4.2.

Theorem 4.4. *Suppose a design $D = (d_{ij})$ has n rows, m columns and entries from $GF(s^p)$, $E(D; y)$ reaches the lower bound in (4.5) if it is balanced and has the properties that for any $a, b \in \{1, \cdots, n\}, a \neq b$, the set $\{d_{aj} - d_{bj}|1 \leq j \leq m\}$ contains every elements of $GF(s^p)$ the same number of times.*

A class of optimal designs can be constructed from generalized Hadamard matrices. A generalized Hadamard matrix $H = (h_{ij})$ over an additive group G is an $n \times n$ square matrix such that for any $1 \leq i < j < n$, the set $\{h_{ik} - h_{jk}|1 \leq k \leq n\}$ contains every element in G for the same number of times. The normalized Hadamard matrix has its first column and first row to be vectors of zeroes. Deleting the first column of the normalized generalized Hadamard matrix yields an optimal design. The next theorem shows how generalized Hadamard matrices can be constructed over $GF(s^p)$.

Theorem 4.5. *The multiplication table M over $GF(s^p)$ is a generalized Hadamard matrix. Furthermore, for $x \in GF(s^p)$, $M + x$ is also a generalized Hadamard matrix.*

We give some examples to illustrate the construction.

Example 4.4. *Table 4.1 contains multiplication tables over $GF(2^3)$ based on two irreducible polynomials $y = x^3 + x + 1$ and $y = x^3 + x^2 + 1$, respectively. Both designs achieve the lower bound of the space-filling pattern enumerator after deleting the first column of zeroes. Based on Theorem 4.3, $S_2(D) = 0$ and $S_{21}(D) = 2^7/8 = 2048$ for both designs.*

0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7		0	1	2	3	4	5	6	7
0	2	4	6	3	1	7	5		0	2	4	6	5	7	1	3
0	3	6	5	7	4	1	2		0	3	6	5	1	2	7	4
0	4	3	7	6	2	5	1		0	4	5	1	7	3	2	6
0	5	1	4	2	7	3	6		0	5	7	2	3	6	4	1
0	6	7	1	5	3	2	4		0	6	1	7	2	4	3	5
0	7	5	2	1	6	4	3		0	7	3	4	6	1	5	2

Table 4.1: Multiplication tables over $GF(2^3)$ based on $y = x^3 + x + 1$ (left) and $y = x^3 + x^2 + 1$ (right).

Example 4.5. *Table 4.2 contains the multiplication table over $GF(2^4)$ based on irreducible polynomials $y = x^4 + x + 1$. We get an optimal design after deleting the first column of zeroes.*

We can construct more generalized Hadamard matrices via collapsing levels. Let M be the multiplication table over $GF(s^p)$. For any $q < p$, we can collapse M into s^q levels. The collapse is done by transforming elements $x = (f_1 \cdots f_p)_s \in GF(s^p)$ to $x' = (f_1 \cdots f_q)_s \in GF(s^q)$ where $(f_1 \cdots f_q)$ is the first q elements of $(f_1 \cdots f_p)$. Table 4.3 presents another optimal design over $GF(2^3)$ when Table 4.2 is collapsed to 8 levels.

Theorem 4.6. *For $q < p$, when the multiplication table M over $GF(s^p)$ is collapsed into s^q levels, the collapsed table is a generalized Hadamard matrix over $GF(s^q)$.*

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	2	4	6	8	10	12	14	3	1	7	5	11	9	15	13
0	3	6	5	12	15	10	9	11	8	13	14	7	4	1	2
0	4	8	12	3	7	11	15	6	2	14	10	5	1	13	9
0	5	10	15	7	2	13	8	14	11	4	1	9	12	3	6
0	6	12	10	11	13	7	1	5	3	9	15	14	8	2	4
0	7	14	9	15	8	1	6	13	10	3	4	2	5	12	11
0	8	3	11	6	14	5	13	12	4	15	7	10	2	9	1
0	9	1	8	2	11	3	10	4	13	5	12	6	15	7	14
0	10	7	13	14	4	9	3	15	5	8	2	1	11	6	12
0	11	5	14	10	1	15	4	7	12	2	9	13	6	8	3
0	12	11	7	5	9	14	2	10	6	1	13	15	3	4	8
0	13	9	4	1	12	8	5	2	15	11	6	3	14	10	7
0	14	15	1	13	3	2	12	9	7	6	8	4	10	11	5
0	15	13	2	9	6	4	11	1	14	12	3	8	7	5	10

Table 4.2: Multiplication table over $GF(2^4)$ based on $y = x^4 + x + 1$.

Theorems 4.5 and 4.6 provides a class of optimal designs that achieve the lower bounds in Theorems 4.1 and 4.2. These optimal designs have $n = s^p$ runs, $m = s^p - 1$ columns and s^p or s^q levels. In the next subsection, we compare these optimal designs with other types of space-filling designs.

4.3.2 Comparisons with Other Space-filling Designs

In the first example, we compare various discrepancies of our optimal designs with other space-filling designs.

Example 4.6. *We randomly generate four types of* 16×15 *designs: uniform design samples (UD),*

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	2	2	3	3	4	4	5	5	6	6	7	7
0	1	2	3	4	5	6	7	1	0	3	2	5	4	7	6
0	1	3	2	6	7	5	4	5	4	6	7	3	2	0	1
0	2	4	6	1	3	5	7	3	1	7	5	2	0	6	4
0	2	5	7	3	1	6	4	7	5	2	0	4	6	1	3
0	3	6	5	5	6	3	0	2	1	4	7	7	4	1	2
0	3	7	4	7	4	0	3	6	5	1	2	1	2	6	5
0	4	1	5	3	7	2	6	6	2	7	3	5	1	4	0
0	4	0	4	1	5	1	5	2	6	2	6	3	7	3	7
0	5	3	6	7	2	4	1	7	2	4	1	0	5	3	6
0	5	2	7	5	0	7	2	3	6	1	4	6	3	4	1
0	6	5	3	2	4	7	1	5	3	0	6	7	1	2	4
0	6	4	2	0	6	4	2	1	7	5	3	1	7	5	3
0	7	7	0	6	1	1	6	4	3	3	4	2	5	5	2
0	7	6	1	4	3	2	5	0	7	6	1	4	3	2	5

Table 4.3: Multiplication table over $GF(2^4)$ collapsed to $GF(2^3)$.

Latin hypercube samples (LHS), maximin Latin hypercube samples (mLHS) and maximum projection Latin hypercube samples (mpLHS) from R packages `UniDOE`, `lhs` and `MaxPro`. For each type of designs, we generate designs 100 times. We obtain an optimal design (OP) from Table 4.2 by deleting the first column. If we add $x \in GF(2^4)$ to all the entries of the optimal design, it is still optimal. Thus, we generate 16 optimal designs by adding each $x \in GF(2^4)$ to the original design. We calculate the centered L_2-discrepancy (CD), the warp-around L_2-discrepancy (WD), the mixture L_2-discrepancy (MD) and the stratified L_2-discrepancy (SD2) with $s = 2, p = 4$ for the five types of designs. The results are in Figure 4.2. Our designs are the best under the stratified L_2-discrepancy and are competitive under other discrepancies. Table 4.4 shows the best discrep-

ancies each type of design samples can reach and the time for generating one design in R with intel core i5 laptop. Uniform designs have the smallest centered L_2-discrepancy and mixture L_2-discrepancy, while our designs have the smallest warp-around L_2-discrepancy. The algorithmic search of uniform designs takes considerable amount of time. Optimal designs, on the other hand, are constructed instantly. When the dimension of the design region gets larger, uniform designs suffer from the curse of dimensionality, while our designs are not affected by the curse of dimensionality.

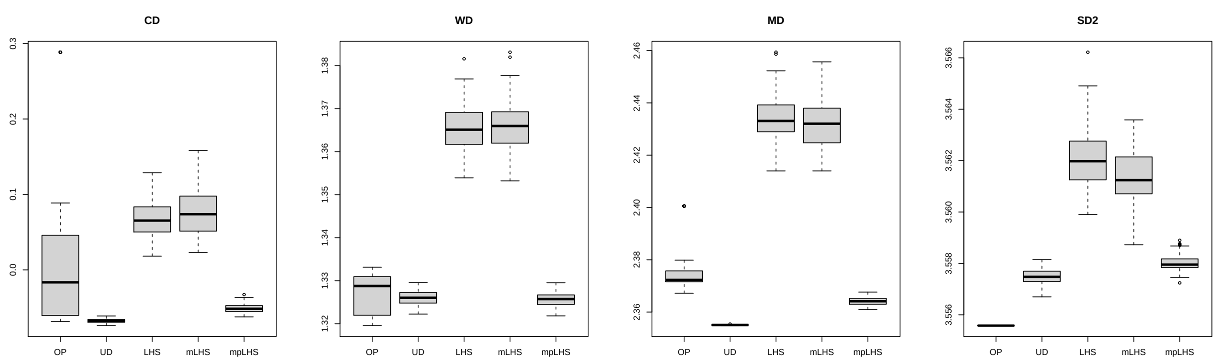

Figure 4.2: Logarithmic discrepancies for five types of designs.

Table 4.4: The best discrepancies for five types of designs and the time used.

Design	CD	WD	MD	SD2	Time (s)
OP	0.9337	**3.7418**	10.6674	**35.0081**	0
UD	**0.9287**	3.7518	**10.5362**	35.0473	19
LHS	1.0183	3.8726	11.1785	35.1597	0
mLHS	1.0234	3.8699	11.1785	35.1184	0
mpLHS	0.9396	3.7503	10.6011	35.0664	1

In the second example, we compare the projection properties of our optimal designs with other space-filling designs. The maximum projection design $D_{n\times m} = (d_{ij})$ proposed by Joseph et al.

(2015) is constructed by minimizing $\psi(D)$ defined in the following:

$$\psi(D) = \left\{ \frac{1}{\binom{n}{2}} \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} \frac{1}{\prod_{j=1}^{m} (d_{aj} - d_{bj})^2} \right\}^{1/m}.$$

The denominator of $\psi(D)$ has products of squared distances from all m factors. In any projections, no two points can get close to each other.

Example 4.7. *We consider four 16×15 Latin hypercube designs and compare the maximum $\psi(D)$ and the minimum Euclidean distance of the designs when they are projected to $3 \leq k \leq 15$ dimensions. We choose the best uniform design, maximin Latin hypercube design and maximum projection Latin hypercube design among the 100 designs according to their respective criteria. From the 16 optimal designs in Example 4.6, we choose the one with the smallest centered L_2-discrepancy. Figure 4.3 shows the values of the maximum $\psi(D)$ and minimum Euclidean distance from all projection designs onto $3 \leq k \leq 15$ dimensions. As expected, for $k = 15$ (i.e., without projection), the maximin Latin hypercube design is the best with the largest minimum Euclidean distance whereas the maximum projection Latin hypercube design is the best with the smallest maximum $\psi(D)$. For $3 \leq k < 15$, either the optimal design or the maximum projection Latin hypercube design is the best. Particularly, the optimal design is the best for k =3–6 and 11–14 dimensions with the largest minimum Euclidean distance and k =3–8 dimensions with the smallest maximum $\psi(D)$. Overall, our design outperforms the rest of the space-filling designs in lower-dimensional projections, which is a natural product of stratification. When the number of active factors is small, the lower-dimensional projection properties are often preferred. From this perspective, our design shows good potential in screening effective factors in computer experiments.*

4.4 Concluding Remarks

This chapter develops lower bounds for the stratified L_2-discrepancy and the space-filling pattern enumerator via defining a metric space that reveals the distance between points based on stratification. Both criteria evaluate the uniformity of the designs based on stratification properties.

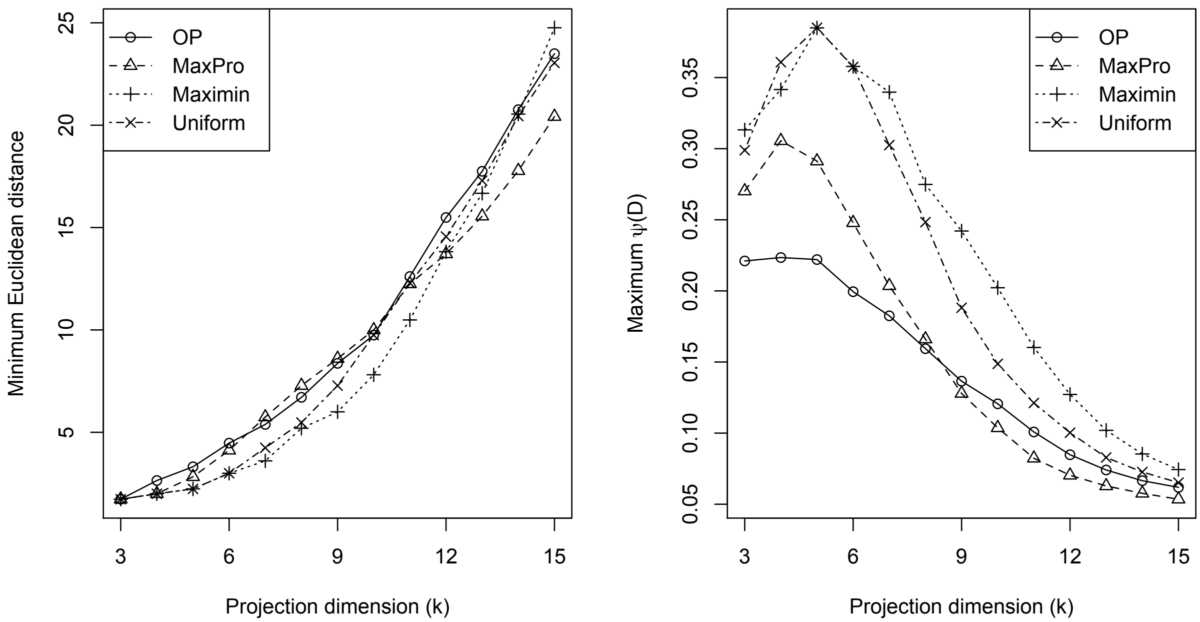

Figure 4.3: Plots of minimum Euclidean distance (the larger the better, on the left) and maximum $\psi(D)$ (the smaller the better, on the right) against k.

The conditions of which designs reach the lower bounds are the same for both criteria. Thus, the optimal designs of the stratified L_2-discrepancy are optimal under the space-filling criterion. One class of designs that achieve the lower bounds are the generalized Hadamard matrices. We present several simple methods for the construction of generalized Hadamard matrices based on multiplication tables on the Galois field. The optimal design we constructed shows competitive performance under other discrepancies. In addition, the optimal design performs best in lower-dimensional projections.

This chapter opens some interesting topics. New construction methods of optimal designs could be developed. One possible direction is to conduct transformation on columns and rows of the generalized Hadamard matrices. The idea of stratification has deep connection with digital nets proposed for quasi-Monte Carlo methods. There are good theoretical results on the variance of Monte Carlo estimates from digital nets. It is interesting to study whether these results are valid in computer experiments when designs that achieve the lower bounds are used. Another interesting topic is to explore the connection between the good lattice point designs and the digital nets.

4.5 Appendix: Proofs

Proof of Theorem 4.1. If $x = (f_1(x) \cdots f_p(x))_s \in \mathbb{Z}_{s^p}$, there are $s^{k-1}(s-1)$ possible $y \in \mathbb{Z}_{s^p}$ that satisfy $\rho(x, y) = k, k = 1, \cdots, p$. Those ys are in set

$$\begin{aligned}\{y = (f_1(y) \cdots, f_p(y))_s | f_i(y) = f_i(x), i = 1, \cdots, p-k,\\ f_{p-k+1}(y) \in \{1, \cdots, s-1\},\\ f_i(y) \in \mathbb{Z}_{s^p}, i = p-k+2, \cdots, p\}.\end{aligned}$$

If $x, y \in \mathbb{Z}_{s^p}$, $\rho(x, y) = 0$ if and only if $x = y$.

For design $\mathcal{P} = (x_{ij})_{n\times m}$ with entries from C, if the elements of each column of $\mathcal{P}$ spread evenly when C is stratified to s^p grids, the elements of each column of design $D = \lfloor s^p\mathcal{P} \rfloor$ becomes a multiset that includes the elements in $\mathbb{Z}_{s^p}$ for λ times. We have

$$\sum_{a=1}^{n}\sum_{b\neq a} n_0(a, b) = (\lambda - 1)nm, \quad \sum_{a=1}^{n}\sum_{b\neq a} n_k(a, b) = \lambda s^{k-1}(s-1)nm.$$

By the arithmetic and geometric means inequality,

$$\frac{1}{n(n-1)}\sum_{a=1}^{n}\sum_{b\neq a}^{n}\prod_{k=0}^{p}\sigma_k^{n_k(a,b)} \geq \left(\prod_{a=1}^{n}\prod_{b\neq a}^{n}\prod_{k=0}^{p}\sigma_k^{n_k(a,b)}\right)^{\frac{1}{n(n-1)}} = \sigma_0^{\frac{(\lambda-1)m}{(n-1)}}\prod_{k=1}^{p}\sigma_k^{\frac{\lambda s^{k-1}(s-1)m}{(n-1)}}.$$

Then the lower bound of the stratified L_2-discrepancy follows from (4.3). □

Proof of Theorem 4.2. Theorem 4.2 follows from Theorem 4.1 and Lemma 4.2. □

Proof of Theorem 4.5. Denote the elements in $GF(s^p)$ by $\{\alpha_0, \cdots, \alpha_{s^p-1}\}$. If M is a multiplication table over $GF(s^p)$, the subtraction between any two rows of M is

$$\begin{pmatrix}\beta\alpha_0\\ \vdots\\ \beta\alpha_{s^p-1}\end{pmatrix} - \begin{pmatrix}\gamma\alpha_0\\ \vdots\\ \gamma\alpha_{s^p-1}\end{pmatrix} = (\beta-\gamma)\begin{pmatrix}\alpha_0\\ \vdots\\ \alpha_{s^p-1}\end{pmatrix},$$

where $\beta, \gamma \in GF(s^p), \beta \neq \gamma$. The elements $(\beta - \gamma)\alpha_i, i = 0 \cdots, s^p - 1$ include every element of $GF(s^p)$ exactly once. Furthermore, adding any $x \in GF(s^p)$ to M will not change this property. This completes the proof. $\square$

Proof of Theorem 4.6. Denote the collapse operation by $g(\cdot)$ and we have

$$g((f_1 \cdots f_p)_s) = (f_1 \cdots f_q)_s, q < p.$$

Using the polynomial representation, for $y_1 = f_1(y_1) + f_2(y_1)x + \cdots + f_p(y_1)x^{p-1}$ and $y_2 = f_1(y_2) + f_2(y_2)x + \cdots + f_p(y_2)x^{p-1}$, we have $g(y_1) = f_1(y_1) + f_2(y_1)x + \cdots + f_q(y_1)x^{q-1}$ and $g(y_2) = f_1(y_2) + f_2(y_2)x + \cdots + f_q(y_2)x^{q-1}$. It is obvious that

$$g(y_1) - g(y_2) = g(y_1 - y_2).$$

The subtraction between any two rows of collapsed M is

$$\begin{pmatrix} g(\beta\alpha_0) \\ \vdots \\ g(\beta\alpha_{s^p-1}) \end{pmatrix} - \begin{pmatrix} g(\gamma\alpha_0) \\ \vdots \\ g(\gamma\alpha_{s^p-1}) \end{pmatrix} = \begin{pmatrix} g((\beta-\gamma)\alpha_0) \\ \vdots \\ g((\beta-\gamma)\alpha_{s^p-1}) \end{pmatrix}.$$

The elements $g((\beta - \gamma)\alpha_i), i = 0 \cdots, s^p - 1, \beta \neq \gamma$ include every element of $GF(s^q)$ exactly s^{p-q} times. $\square$

CHAPTER 5

Conclusion

Computer models are computer codes used to simulate complicated, hard-to-solve systems. Computer experiments aim to build statistical surrogate models efficiently based on the data from computer models. Space-filling designs are widely used in computer experiments. Inspired by the stratified orthogonality of the strong orthogonal arrays, we concentrate on creating criteria to fully characterize the space-filling properties based on stratification.

A minimum aberration type criterion is proposed in Chapter 2 as a systematic way of classifying and ranking space-filling designs including various types of strong orthogonal arrays and Latin hypercube designs. Space-filling hierarchy principle is proposed as the basic assumption of the space-filling criterion. The idea of the space-filling hierarchy principle generalizes the strength of strong orthogonal arrays. The space-filling pattern, analogous to the wordlength pattern, reveals various stratification properties of the design clustered by the volume of the grids. We select designs that sequentially minimize the elements of the space-filling pattern. Strong orthogonal arrays of maximum strength are favorable under the proposed criterion. Values of the space-filling pattern can further rank strong orthogonal arrays with the same strength.

One of the drawbacks of the space-filling criterion is that it is restricted to designs with a fixed number of levels. We propose the stratified L_2-discrepancy for evaluating space-filling properties of designs based on design stratification properties. The new discrepancy is suitable for evaluating all kinds of designs with little curse of dimensionality. Using the framework of reproducing kernel, the stratified L_2-discrepancy inherits most of the advantages of the generalized L_2-discrepancy. The stratification scheme and the importance levels of each stratification can be tuned flexibly. The stratified L_2-discrepancy is more general and easier to compute than the space-filling criterion.

The proposal of the space-filling criterion and the stratified L_2-discrepancy calls for construction of optimal designs. We derive lower bounds for the stratified L_2-discrepancy and the space-filling pattern enumerator via defining a metric space that reveals the distance between points based on stratification. The lower bounds are not related to the weights allocated to each stratification. As a result, the optimal designs for the stratified L_2-discrepancy are optimal for the space-filling criterion. The generalized Hadamard matrices are proved to be a class of optimal designs that achieve the lower bounds. Simple construction methods of the generalized Hadamard matrices are proposed based on Galois fields. The algorithmic search of uniform designs sometimes takes considerable amount of time. Our optimal designs for the stratified L_2-discrepancy, on the other hand, are constructed instantly. Comparisons between the optimal designs and other space-filling designs show that optimal designs are space-filling in lower-dimensional projections.

Bibliography

Ba, S., Myers, W. R., and Brenneman, W. A. (2015), "Optimal sliced Latin hypercube designs," *Technometrics*, 57, 479–487.

Bierbrauer, J., Edel, Y., and Schmid, W. C. (2002), "Coding-theoretic constructions for (t, m, s)-nets and ordered orthogonal arrays," *Journal of Combinatorial Designs*, 10, 403–418.

Chen, H., Loeppky, J. L., Sacks, J., and Welch, W. J. (2016), "Analysis methods for computer experiments: How to assess and what counts?" *Statistical Science*, 31, 40–60.

Cheng, C.-S. (2014), *Theory of Factorial Design: Single-and Multi-Stratum Experiments*, CRC Press.

Fang, K.-T., Li, R., and Sudjianto, A. (2006), *Design and Modeling for Computer Experiments*, Chapman and Hall/CRC.

Fang, K.-T., Liu, M.-Q., Qin, H., and Zhou, Y.-D. (2018), *Theory and Application of Uniform Experimental Designs*, Springer.

He, L., Qin, H., and Ning, J. (2020), "Weighted symmetrized centered discrepancy for uniform design," *Communications in Statistics-Simulation and Computation*, 0, 1–11.

He, X. (2020), "Lattice-based designs with quasi-optimal separation distance on all projections," *Biometrika*, asaa057.

He, Y., Cheng, C.-S., and Tang, B. (2018), "Strong orthogonal arrays of strength two plus," *The Annals of Statistics*, 46, 457–468.

He, Y. and Tang, B. (2013), "Strong orthogonal arrays and associated Latin hypercubes for computer experiments," *Biometrika*, 100, 254–260.

— (2014), "A characterization of strong orthogonal arrays of strength three," *The Annals of Statistics*, 42, 1347 – 1360.

Hickernell, F. (1998), "A generalized discrepancy and quadrature error bound," *Mathematics of Computation*, 67, 299–322.

Hua, L.-K. and Wang, Y. (1981), *Applications of Number Theory to Numerical Analysis*, Springer Science & Business Media.

Joseph, V. R., Gul, E., and Ba, S. (2015), "Maximum projection designs for computer experiments," *Biometrika*, 102, 371–380.

Li, W., Liu, M.-Q., and Tang, B. (2020), "A method of constructing maximin distance designs," *Biometrika*, asaa089.

Li, W., Liu, M.-Q., and Yang, J.-F. (2021), "Column-orthogonal nearly strong orthogonal arrays," *Journal of Statistical Planning and Inference*, 215, 184–192.

Lin, C. D. and Tang, B. (2015), "Latin hypercubes and space-filling designs," in *Handbook of Design and Analysis of Experiments*, eds. Dean, A., Morris, M., Stufken, J., and Bingham, D., Chapman and Hall/CRC, pp. 593–625.

Martin, W. and Stinson, D. (1999), "Association schemes for ordered orthogonal arrays and (T, M, S)-nets," *Canadian Journal of Mathematics*, 51, 326–346.

Mukerjee, R. and Wu, C. F. J. (2006), *A Modern Theory of Factorial Design*, Springer.

Mullen, G. L. and Schmid, W. C. (1996), "An equivalence between (t, m, s)-nets and strongly orthogonal hypercubes," *Journal of Combinatorial Theory, Series A*, 76, 164–174.

Niederreiter, H. (1986), "Low-discrepancy point sets," *Monatshefte für Mathematik*, 102, 155–167.

— (1987a), "Point sets and sequences with small discrepancy," *Monatshefte für Mathematik*, 104, 273–337.

— (1987b), "A statistical analysis of generalized feedback shift register pseudorandom number generators," *SIAM Journal on Scientific and Statistical Computing*, 8, 1035–1051.

Owen, A. B. (1992), "Orthogonal arrays for computer experiments, integration and visualization," *Statistica Sinica*, 2, 439–452.

Rosenbloom, M. Y. and Tsfasman, M. A. (1997), "Codes for the m-metric," *Problemy Peredachi Informatsii*, 33, 55–63.

Santner, T. J., Williams, B. J., and Notz, W. (2003), *The Design and Analysis of Computer Experiments*, Springer.

Schmid, W. C. (1995), "(t,m,s)-Nets: Digital Construction and Combinatorial Aspects," Ph.D. thesis, University Salzburg.

Shi, C. and Tang, B. (2019), "Design selection for strong orthogonal arrays," *Canadian Journal of Statistics*, 47, 302–314.

— (2020), "Construction results for strong orthogonal arrays of strength three," *Bernoulli*, 26, 418–431.

Sun, F., Wang, Y., and Xu, H. (2019), "Uniform projection designs," *The Annals of Statistics*, 47, 641–661.

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

— (2001), "Theory of J-characteristics for fractional factorial designs and projection justification of minimum G_2-aberration," *Biometrika*, 88, 401–407.

Tang, B. and Deng, L.-Y. (1999), "Minimum G_2-aberration for nonregular fractional factorial designs," *The Annals of Statistics*, 27, 1914–1926.

Tang, Y., Xu, H., and Lin, D. K. J. (2012), "Uniform fractional factorial designs," *The Annals of Statistics*, 40, 891–907.

Tian, Y. and Xu, H. (2021), "A Minimum Aberration Type Criterion for Selecting Space-Filling Designs," *Biometrika*, asab021.

Wang, L., Xiao, Q., and Xu, H. (2018), "Optimal maximin L_1-distance Latin hypercube designs based on good lattice point designs," *The Annals of Statistics*, 46, 3741–3766.

Wang, Y., Sun, F., and Xu, H. (2020), "On Design Orthogonality, Maximin Distance, and Projection Uniformity for Computer Experiments," *Journal of the American Statistical Association*, 1–11.

Weyl, H. (1916), "Über die gleichverteilung von zahlen mod. eins," *Mathematische Annalen*, 77, 313–352.

Winker, P. and Fang, K.-T. (1997), "Application of threshold-accepting to the evaluation of the discrepancy of a set of points," *SIAM Journal on Numerical Analysis*, 34, 2028–2042.

Wu, C. F. J. and Hamada, M. S. (2009), *Experiments: Planning, Analysis, and Optimization*, John Wiley & Sons, 2nd ed.

Xiao, Q. and Xu, H. (2017), "Construction of maximin distance Latin squares and related Latin hypercube designs," *Biometrika*, 104, 455–464.

— (2018), "Construction of maximin distance designs via level permutation and expansion," *Statistica Sinica*, 28, 1395–1414.

Xu, H. and Wu, C. F. J. (2001), "Generalized minimum aberration for asymmetrical fractional factorial designs," *The Annals of Statistics*, 29, 1066–1077.

Zhang, A., Li, H., Quan, S., and Yang, Z. (2018), "UniDOE: Uniform design of experiments," *R package version*, 1.

Zhou, Y. and Tang, B. (2019), "Column-orthogonal strong orthogonal arrays of strength two plus and three minus," *Biometrika*, 106, 997–1004.

Zhou, Y. and Xu, H. (2015), “Space-filling properties of good lattice point sets,” *Biometrika*, 102, 959–966.

Zhou, Y.-D., Fang, K.-T., and Ning, J.-H. (2013), “Mixture discrepancy for quasi-random point sets,” *Journal of Complexity*, 29, 283–301.

Zhou, Y.-D. and Xu, H. (2014), “Space-filling fractional factorial designs,” *Journal of the American Statistical Association*, 109, 1134–1144.