Nearly Orthogonal Arrays of Strength Three

by

Thellamorage Dona Darsha Kalpani Perera

B.Sc., University of Sri Jayewardenepura, 2019

Project Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
Department of Statistics and Actuarial Science
Faculty of Science

SIMON FRASER UNIVERSITY
Summer 2023

Declaration of Committee

Name: **Thellamorage Dona Darsha Kalpani Perera**

Degree: **Master of Science**

Thesis title: **Nearly Orthogonal Arrays of Strength Three**

Committee:

Chair: Liangliang Wang
Associate Professor, Statistics and Actuarial Science

Boxin Tang
Supervisor
Professor, Statistics and Actuarial Science

Tom Loughin
Committee Member
Professor, Statistics and Actuarial Science

Himchan Jeong
Examiner
Assistant Professor, Statistics and Actuarial Science

Abstract

The main effects of a factorial experiment can be estimated with minimum variance and zero bias using orthogonal arrays (OAs) of strength three. However, such arrays require the run size to be a multiple of eight. When the run size is not a multiple of eight, OAs of strength three do not exist. In the presence of non-negligible two-factor interactions, OAs of strength two with minimum G_2-aberration are available as variance-optimal designs that have the minimum bias among non-isomorphic OAs of strength two. Such designs only require the run size to be a multiple of four. Best fold-over designs have zero bias and provide the minimum variance among all fold-over designs. We examine the use of nearly orthogonal arrays of strength three for estimating main effects in the presence of nonnegligible two-factor interactions. This provides an approach that is capable of balancing the consideration of variance and bias. Our method is compared with the two existing classes of designs.

Keywords: Effect sparsity; Fold-over design; Mean squared error; Projection property

Dedication

To my dearest husband, Viraj for his unconditional support, encouragement and motivation.

Acknowledgements

First of all I would like to express my deepest gratitude to my senior supervisor, Prof. Boxin Tang for his patient guidance, great supervision, immense support and kind encouragement over the past years. I was able to complete the research within the timeline only because of his proper mentorship. This would not have been possible without his insights, thoughts, suggestions and clear guidance. Weekly meetings with him let me gather a massive amount of knowledge and experience. I am forever grateful to him for giving me an opportunity to work under his great supervision. A big thank to my colleges at SFU for always sharing their thoughts on whatever I do. The moments we spend and the memories we create together would forever be cherished. I would also like to thank all the professors at the department for helping me to learn subject-related knowledge as well as the life lessons given. A special gratitude to all the professors who taught me in my master's degree including Dr. Joan Hu, Dr. Rachel Altman, Dr. Richard Lockhart, Dr. Jinko Graham, Ian Bercovitz and Dr. Boxin Tang. I am thankful to my department at SFU for providing all the resources, facilities and creating a friendly environment for students. I would also like to mention my husband, the biggest support system that I have in my life, for always believing in me and encouraging me to achieve the best at every step I take forward. Last but not least, my family in Sri Lanka, mom, dad and my only brother for making who I am today.

Table of Contents

Declaration of Committee **ii**

Abstract **iii**

Dedication **iv**

Acknowledgements **v**

Table of Contents **vi**

List of Tables **viii**

List of Figures **ix**

1 Introduction **1**

1.1 Overview and the Motivation of the Study 1

1.2 Fractional Factorial Designs 3

1.3 J-Characteristic and a Measure of Three-Dimensional Projection Property . 6

1.4 Bias, Variance, and Mean Squared Error 7

2 Design Methodology **10**

2.1 Variance-Optimal Designs 10

2.2 Bias-Optimal Designs 13

2.3 Nearly Orthogonal Arrays of Strength Three and Their Construction . . . 14

2.3.1 Partial Fold-Over of OAs of Strength Two 14

2.3.2 Adding Runs to OAs of Strength Two 15

2.3.3 Local Search Algorithm 15

2.4 An MSE Criterion for Comparing Different Designs 17

3 Results of Comparisons **19**

3.1 Comparison of Variance-Optimal Designs and Bias-Optimal Designs with Best Nearly Orthogonal Arrays of Strength Three 21

3.2 Comparison of Variance-Optimal Designs and Best Nearly Orthogonal Arrays of Strength Three When Best Fold-Over Designs Do Not Exist 27

4 Concluding Remarks **31**

Bibliography **32**

Appendix A List of Designs **33**

A.1 Designs used in the case where $m = n/2$; the Variance-Optimal Design, Bias-Optimal Design and the Best NOA respectively. 33

A.2 Designs used in the case where $m = n/2 + 1$; the Variance-Optimal Design and the Best NOA respectively. 36

List of Tables

Table 1.1 Main Effects and Interaction Effects of a Five-Factor Experiment . . 3

Table 2.1 Complete Set of Iterations of the Local Search Algorithm For the 20×10 Experiment 17

Table 3.1 Variance, Bias and MSE for Variance and Bias-Optimal Designs with 10 Runs and 5 Factors When All Two-Factor Interactions are Significant ($\pi = 1$) 20

Table 3.2 Three Dimensional Projection Property of Variance-Optimal Designs, Bias-Optimal Designs and Best NOAs in experiments with n=10, 12, 14, 18, 20, 28 where $m = n/2$ 22

Table 3.3 C_1 and C_2 at different values of π in experiments with $m = n/2$ . . . 26

Table 3.4 Three Dimensional Projection Property of Variance-Optimal Designs and Best NOAs in Experiments Where $m = n/2 + 1$ 27

Table 3.5 Comparison of the MSE for the Variance-Optimal Design and the Best NOA With 10 Runs and 6 Factors When All Two-Factor Interactions are Significant 28

Table 3.6 C^* at different values of π in experiments with $m = n/2 + 1$ 30

List of Figures

Figure 2.1 First Two Steps of the Local Search Algorithm 16

Figure 3.1 Comparison of the MSE for Variance and Bias-Optimal Designs with n=10 and m=5 when $\pi = 1$ 21

Figure 3.2 Comparison of the MSE for Designs With n=10 and m=5 24

Figure 3.3 C_1 and C_2 of the experiment of 10×5 as a function of π 26

Figure 3.4 Comparison of the MSE of Designs with n=10 and m=6 29

Figure 3.5 C^* of the experiment of 10×6 as a function of π 30

Chapter 1

Introduction

1.1 Overview and the Motivation of the Study

Design of experiments is a branch of applied statistics, where the data are collected and analyzed to evaluate the effect of a set of factors on a variable of interest, called a response variable. While there are different types of design of experiments that are commonly used in the industry, our study focuses on factorial designs. Factorial experiments consist of two or more factors, each having a specified number of levels. The designs used in our study have two levels that are coded as ± 1 for the ease of handling, where -1 refers to the low level and $+1$ refers to the high level. Factors in such experiments produce two different effects, namely, main effects and interaction effects. The main effect of a factor quantifies its effect on the response variable averaged across the other factors. Interaction effects occur when the effect of a factor on the response variable is influenced by one or more other factors being considered in the study. There are $2^m - 1$ effects in total in a factorial experiment with m two-level factors.

A full factorial is a design that uses all possible combinations of all the factors. The runs of such a design refer to the 2^m treatment combinations. For instance, a full factorial design with three factors each having two levels is called a $2 \times 2 \times 2$ or a 2^3 design containing $2^3 = 8$ distinct treatment combinations. At least 8 runs are required to estimate all seven effects. It is clear that the number of runs required to estimate all the main effects and the interaction effects in a full factorial experiment increases exponentially with the number of factors, which renders full factorials impractical. Fractional factorial designs provide a remedy for the expensive nature of full factorial designs. As the name itself suggests, such a design contains only a fraction of the complete set of runs required by its underlying full factorial design. All the designs used in our study are fractional factorials.

We consider the problem of estimating main effects in the presence of two-factor interactions using fractional factorial designs. Orthogonal arrays of strength three allow the estimation of the main effects of a factorial experiment with minimum variance and zero bias even when some two factor interactions are non-negligible. However, such arrays require the run size to be a multiple of eight. OAs of strength two and fold-over designs are two existing classes of designs for estimating the main effects when OAs of strength three are unavailable. OAs of strength two are variance-optimal and only require the run size to be a multiple of four. On the other hand, non-orthogonal fold-over designs introduced by Margolin (1969) have zero bias. In our study, we use the OAs of strength two with minimum G_2-aberration as variance-optimal designs, and such OAs provide the minimum bias among the same order non-isomorphic OAs (Tang and Deng (1999)). Non-orthogonal fold-over designs with the lowest variance are bias-optimal, and they are called the best fold-over designs (BFDs). Our goal is to introduce and study a class of alternative designs, which outperform the two existing classes of designs by balancing the bias-variance trade off.

In this chapter, we provide a brief discussion on fractional factorial designs, including orthogonal arrays, the concept of J-characteristic and the three dimensional projection property, and finally the bias, variance, and the mean squared error of linear models. We review the two existing classes of designs used in our study, namely variance-optimal designs and bias-optimal designs in Chapter 2. Further, we introduce an alternative set of designs called nearly orthogonal arrays of strength three and study different approaches for their construction. We end the chapter with an overview of the calculation of the mean squared error by taking effect sparsity into consideration, which will be used to compare the three types of designs later on. Chapter 3 summarizes the important results we obtained from our empirical study. We provide a summary and conclusions in Chapter 4.

1.2 Fractional Factorial Designs

Fractional factorial designs are useful in examining the effects of a large number of factors on a response variable of interest using a relatively small number of experimental runs. Any experimental design that contains less than 2^m runs in an m-factor experiment is a fractional factorial. Regular fractional factorial designs are referred to as 2^{m-p} designs, where the p indicates the corresponding fraction of runs.

Consider a five-factor design each with two levels. A full factorial design requires $2^5 = 32$ runs to estimate all its effects as mentioned below.

		Interactions			
Average	Main Effects	2-factor	3-factor	4-factor	5-factor
1	5	10	10	5	1

Table 1.1: Main Effects and Interaction Effects of a Five-Factor Experiment

However, many of these effects might be negligible in practice. In general, higher order interactions are not of our interest. Very often, experimenters' main interest lies in estimating main effects, and hence, it is meaningless to carry out all 32 runs to estimate the thirty-one effects in the above example. Moreover, when an experiment involves a large number of factors, higher order interactions are often negligible. That's where fractional factorial designs come into play. For instance, a half fractional factorial design of a five-factor two-level experiment is a 2^{5-1} design. This design uses only sixteen runs to study five factors. Nevertheless, the significant results obtained from both designs will probably be similar (Dey and Mukerjee (1999)).

Regular fractional factorials are constructed using generators. For instance, in the above five-factor half fractional factorial experiment, a full 2^4 design is written for the first four variables; 1, 2, 3 and 4, which fulfills the 16 runs requirement. Then the fifth variable is constructed using the four factor interaction of those variables. Thus, we use the generator 5=1234 or in other words I=12345. The defining relation includes all words that are equal to the identity I, and the words that can be obtained by multiplying all the generators together. As the above design contains only one generator, the defining relation includes only one word, 12345. Suppose we want to construct a 2^{7-4} design. We start with a full 2^3 design written for the first three variables and the rest of the variables are constructed by using the generators 4=12, 5=13, 6=23 and 7=123. The defining relation of such a design is indicated as below:

$$
\begin{aligned}
I &= 124 = 135 = 236 = 1237 = 2345 = 1346 = 347 = 1256 \\
&= 257 = 167 = 456 = 1457 = 2467 = 3567 = 1234567.
\end{aligned}
$$

The effects of regular fractional factorial designs are either orthogonal or fully confounded with each other. Hence, it is not possible to estimate all the effects independently as in full factorial designs.

Orthogonal arrays are a class of fractional factorial designs which include both regular and non-regular designs.

Definition 1. A factorial design of n runs for m factors, each having two-levels, is said to be an orthogonal array of strength t if for any of its sub-matrices of t columns ($n \times t$), the 2^t level combinations occur with the same frequency in the rows. Such an array is denoted by $\mathrm{OA}(n, 2^m, t)$ and the t is called the strength of the OA. If the design is an $\mathrm{OA}(n, 2^m, t)$, then n must be a multiple of 2^t, meaning that $n = \lambda 2^t$ for some positive integer λ (Hedayat et al. (1999)).

The above definition implies that if a design is an OA of strength t, it must also be an OA of any strength $t' < t$. OAs are universally optimal, hence A-optimal. An OA of 8 runs with 4 factors can be displayed as below:

$$\begin{pmatrix} + & + & + & + \\ + & + & - & - \\ + & - & + & - \\ + & - & - & + \\ - & + & + & - \\ - & + & - & + \\ - & - & + & + \\ - & - & - & - \end{pmatrix}$$

In the above design, each t-tuple occurs in the rows with the same frequency for every t=1, 2 and 3, and the run size n is a multiple of $2^3 = 8$. Hence, the design has strength three, and is denoted by $\mathrm{OA}(8, 2^4, 3)$. If a column is removed, the remaining design is still an orthogonal array with strength three. Any two columns in an OA are orthogonal to each other indicating the orthogonality of main effects. In general, OAs of strength three are universally optimal in estimating main effects under a model containing both main effects and two-factor interactions. They are both variance and bias-optimal, providing the minimum variance and zero bias. However, such arrays require the run size to be a multiple of $2^3 = 8$ as explained in Definition 1, which is not always possible under different experimental conditions. In this study, we use OAs of strength two as variance-optimal designs, for the run sizes 12, 20 and 28, where OAs of strength three are not available. OAs of strength two cover a wide range of run sizes as they only require the run size to be a multiple of four.

1.3 J-Characteristic and a Measure of Three-Dimensional Projection Property

J-characteristics are useful indicators to recognize the unique features of factorial designs. The J-characteristic of a single column x is expressed as $J(x)$, which gives the sum of the elements in that particular vector. The J-characteristic $J(x_1, x_2)$ represents the inner product of the two vectors, x_1 and x_2. All the properties of OAs are also fully determined by their J-characteristics. If all the J-characteristics of a factorial design for all possible subsets with t or less columns are zero, the design is an OA with strength t. Consider the design given in the above matrix with eight runs and four factors. If its columns are denoted by x_i, where i=1, 2, 3 and 4, we obtain that

$$J_1 = \{J(x_1), J(x_2), J(x_3), J(x_4)\} = \{0, 0, 0, 0\},$$
$$J_2 = \{J(x_1, x_2), J(x_1, x_3), J(x_2, x_3), J(x_1, x_4), J(x_2, x_4), J(x_3, x_4)\} = \{0, 0, 0, 0, 0, 0\},$$
$$J_3 = \{J(x_1, x_2, x_3), J(x_1, x_2, x_4), J(x_1, x_3, x_4), J(x_2, x_3, x_4)\} = \{0, 0, 0, 0\},$$
$$J_4 = \{J(x_1, x_2, x_3, x_4)\} = 8 = n.$$

J-characteristics being zero for all the subsets containing up to three columns indicates that the design has strength three. A factorial design is called regular if any two factorial effects are either orthogonal or fully aliased with each other. The design becomes non-regular if some of the factorial effects are partially aliased. In regular designs, all its J-characteristics are either 0 or $\pm n$, and it is non-regular if at least one of its characteristics is strictly between 0 and n. The J-characteristics of all possible nonempty subsets are zero in a full factorial or several replicates thereof. Further, if a factorial design is a fold-over design, then the components of the J-characteristics for odd numbers of columns are zero, meaning that $J(x_i) = 0$ and $J(x_i, x_j, x_k) = 0$ for all $i, j, k = 1, 2, 3$ with $i \neq j \neq k$ of a design with only three factors.

The three dimensional projection property (V_3) provides a goodness measure for how close a factorial design is to an OA of strength three. It is a criterion to recognize an optimal design by taking its J_1, J_2 and J_3 characteristics into consideration. Tang (2001) obtained an expression for the three dimensional projection property for a factorial design D, as given below:

$$V_3(D) = 2^{-6} \sum_{j=1}^{3} \binom{m-j}{3-j} \sum_{|t|=j} J_t^2, \tag{1.1}$$

where m is the total number of factors of the design D, and $|t|$ denotes the number of columns in a subset t of columns. Let

$$J_1\text{-Component} = 2^{-6}\tbinom{m-1}{2} \textstyle\sum_{|t|=1} J_t^2,$$
$$J_2\text{-Component} = 2^{-6}\tbinom{m-2}{1} \textstyle\sum_{|t|=2} J_t^2,$$
$$J_3\text{-Component} = 2^{-6} \textstyle\sum_{|t|=3} J_t^2.$$

Equation (1.1) clearly explains that the V_3 is an overall measurement of J_1-, J_2- and J_3-components of a design. As discussed in Section 1.4, J_1 and J_2 characteristics of OAs of strength two are zero, and hence its V_3 measurement completely depends on the J_3-component. On the other hand the V_3 measurement of BFDs in our study are determined by considering only the J_2-component, as the corresponding J_1 and J_3 components are zero in fold-over designs as explained in the previous section. In general, a higher J_2 gives rise to a high variance whereas the bias of a design increases with the J_3. This relationship clearly explains why the OAs of strength two are variance-optimal and BFDs are bias-optimal. In conclusion, designs with the lowest V_3 are considered to be the best.

1.4 Bias, Variance, and Mean Squared Error

Our goal is to introduce a class of optimal designs with respect to some statistical criteria. We consider the concept of A-optimality in our study, which minimizes the sum of the variances of estimated main effects. According to the A-optimality, the design that minimizes the sum of the variances is the optimal design out of the set of designs being compared. Suppose, a sample with n observations are collected, which gives the response vector $Y = (Y_1, Y_2, \ldots, Y_n)^T$. The design matrix with n runs and m factors corresponds to the fractional factorial design being considered in the experiment, which is expressed as

$D = [X_1 \; X_2 \; .... \; X_m]_{n \times m}$. The main-effect linear model indicates the relationship between Y and the set of explanatory variables and is formulated as below:

$$Y = X_{(1)}\beta^{(1)} + \epsilon \tag{1.2}$$

where, ϵ_i's are independently and identically distributed with mean zero and common variance σ^2 for all $i = 1, 2, ..., n$, and $X_{(1)} = [1_n, D]$ is a $n \times (m+1)$ matrix with 1_n being a vector of n ones and $\beta^{(1)} = (\beta_0, \beta_1, \ldots, \beta_m)^T$ being a vector of the grand mean and the main effects of m predictors. The method of least squares estimation allows the unbiased estimation of β's when model (1.2) is correct by minimizing the sum of squares of the residuals, $||Y - X_{(1)}\beta^{(1)}||^2$. Thus, $\hat{\beta}^{(1)} = (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T Y$ is an unbiased estimator for $\beta^{(1)}$ as shown below:

$$\begin{aligned} E(\hat{\beta}^{(1)}) &= (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T E(Y) \\ &= \beta^{(1)}. \end{aligned}$$

The variance covariance matrix of $\hat{\beta}^{(1)}$ is

$$\begin{aligned} Var(\hat{\beta}^{(1)}) &= Var((X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T Y) \\ &= (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T Var(Y) X_{(1)} (X_{(1)}^T X_{(1)})^{-1} \\ &= \sigma^2 (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T X_{(1)} (X_{(1)}^T X_{(1)})^{-1} \\ &= \sigma^2 (X_{(1)}^T X_{(1)})^{-1}. \end{aligned}$$

As σ^2 is a constant, the design that minimizes the $trace(X_{(1)}^T X_{(1)})^{-1}$ is A-optimal. In our study, we allow some two-factor interactions to be non-negligible. The model matrix is then given by $[X_{(1)}, X_{(2)}]$, where $X_{(2)}$ represents the matrix containing all the two-factor interactions of m factors. Let $\beta^{(2)}$ denote the vector of all two-factor interactions. Then the linear model representing the true structure becomes

$$Y = X_{(1)}\beta^{(1)} + X_{(2)}\beta^{(2)} + \epsilon. \tag{1.3}$$

Still our main focus lies on estimating $\beta^{(1)}$ as two-factor interactions are not of our interest. However, $\hat{\beta}^{(1)}$ is no longer unbiased under the model indicated in equation (1.3) as proved below:

$$\begin{aligned}
E(\hat{\beta}^{(1)}) &= (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T E(Y) \\
&= (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T (X_{(1)}\beta^{(1)} + X_{(2)}\beta^{(2)}) \\
&= \beta^{(1)} + (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T X_{(2)}\beta^{(2)},
\end{aligned}$$

$$\begin{aligned}
Bias(\hat{\beta}^{(1)}, \beta^{(1)}) &= E(\hat{\beta}^{(1)}) - \beta^{(1)} \\
&= (X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T X_{(2)}\beta^{(2)}.
\end{aligned}$$

Nevertheless, the variance-covariance matrix of $\hat{\beta}^{(1)}$ is not affected by the two-factor interactions and hence remains unchanged in both models. However, as the estimator is biased under the new model, the MSE will be changed accordingly. The design which gives the minimum MSE is considered the best out of the set of designs being compared, where

$$MSE = \sigma^2 trace(X_{(1)}^T X_{(1)})^{-1} + ||(X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T X_{(2)}\beta^{(2)}||^2. \tag{1.4}$$

All the above models take the intercept into consideration, while our main focus is to estimate main effects. Let $Var^*(\hat{\beta}^{(1)}) = \sigma^2 trace(M)$ with M being the matrix obtained by deleting the first row and the first column of $(X_{(1)}^T X_{(1)})^{-1}$, and $Bias^*(\hat{\beta}^{(1)}, \beta^{(1)}) = B\beta^{(2)}$ with B being the matrix obtained by deleting the first row of $(X_{(1)}^T X_{(1)})^{-1} X_{(1)}^T X_{(2)}$. Now under the model (1.3), the MSE for the main effects becomes:

$$\begin{aligned}
MSE^* &= Var^*(\hat{\beta}^{(1)}) + ||Bias^*(\hat{\beta}^{(1)}, \beta^{(1)})||^2 && (1.5) \\
&= \sigma^2 trace(M) + ||B\beta^{(2)}||^2. && (1.6)
\end{aligned}$$

Chapter 2

Design Methodology

In this chapter, we discuss three classes of designs used in our study. As mentioned before, our main goal is to give an alternative class of designs to use in the situations where OAs of strength three do not exist. Section 2.1 describes variance-optimal designs used in our study, whereas bias-optimal designs are given in Section 2.2. We introduce an alternative class of designs called nearly orthogonal arrays of strength three in Section 2.3 and discuss different approaches used to construct them. Finally, the method used to compare the three types of designs will be discussed in Section 2.4.

2.1 Variance-Optimal Designs

In this study, we consider designs with run sizes 10, 12, 14, 18, 20 and 28. For run sizes 12, 20 and 28, OAs of strength two are available. The OAs used in this study were constructed from Hadamard matrices. A Hadamard matrix is a square matrix with entries +1 and −1, whose rows are mutually orthogonal. This was initially introduced by Sylvester (1867) and later considered by Hadamard (1893). It has the mathematical property that $H^T H = HH^T = nI$, where I is an identity matrix with order n. Hadamard matrices are available for orders 1, 2 and for orders that are multiples of four. The method of tensor product allows the construction of Hadamard matrices with large orders from those with smaller orders. For example, a 4×4 Hadamard matrix can be constructed by a 2×2 Hadarmard matrix as illustrated below.

$$
\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 \times \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} & 1 \times \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \\ \\ 1 \times \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} & -1 \times \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}
$$

A Hadamard matrix is a valuable tool to construct OAs of strength two and three. Consider a Hadamard matrix with the first column containing only 1's, which is said to be normalized. Strength two OAs can be obtained by deleting the all 1's column of the corresponding Hadamard matrix. The resulting OA is denoted by $\mathrm{OA}(n, 2^{n-1}, 2)$, which contains the maximum number of factors to be included in a strength-two OA with n runs. The array obtained by removing one or more columns is still orthogonal with strength two. Hence, a Hadamard matrix with order n can easily be used to construct an $\mathrm{OA}(n, 2^{n'}, 2)$ for any $n' \leq n - 1$.

We choose OAs of strength two with minimum G_2-aberration to be the variance-optimal designs, as they provide the minimum bias among all the non-isomorphic OAs of strength two, while minimizing the variance of estimated main effects among all possible designs. Let $s = \{x_1, ..., x_k\}$ represent any k subset of the design matrix D. Let

$$
B_k(D) = \sum_{|s|=k} (J_k(s)/n)^2,
$$

According to Tang and Deng (1999) and Schoen et al. (2017), for given designs D_1 and D_2, D_1 is said to have less G_2-aberration than D_2 if $B_r(D_1) < B_r(D_2)$, where r is the smallest integer such that $B_r(D_1) \neq B_r(D_2)$. The design D_1 is said to have minimum G_2-aberration if no other design has less G_2-aberration than D_1. A minimum G_2-aberration design estimates the main effects with minimum bias when compared to all other OAs of strength two.

We also consider run sizes 10, 14 and 18 in our study, which involve non-orthogonal fractional factorial designs. According to Dey and Mukerjee (1999), when the run size of a design is even but not a multiple of four, the variance-optimal designs can be obtained by adding two specific runs to the closest lower order OA.

Lemma 1. The design obtained by adding two runs of form $l_1 = (1, ..., 1)$ and $l_2 = (1, ..., 1, -1, ..., -1)$ to an orthogonal array is universally optimal and hence A-optimal.

Suppose a factorial design consists of m factors. Then l_1 is a vector containing m +1's and l_2 contains m_1 and m_2 +1's and -1's respectively, whereas m_1 is the largest integer that is smaller than or equal to $m/2$ and $m_2 = m - m_1$. To illustrate, we obtain a variance-optimal design for 10 runs with 5 factors by using five columns of the Hadamard matrix of order 8 and then adding two specific runs with l_1 being (1,1,1,1,1) and l_2 being (1,1,-1,-1,-1) to it. In short, while we use OAs of strength two for run sizes 12, 20 and 28, Lemma 1 is used for run sizes 10, 14 and 18 in order to obtain the variance-optimal designs. The below 10×5 variance-optimal design is obtained by considering the first five columns starting from the second column of the Hadamard matrix of order eight.

$$\left(\begin{array}{rrrrr}
1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 \\
-1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & 1 \\
\hline
1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & -1 & -1
\end{array}\right)$$

2.2 Bias-Optimal Designs

Even though variance-optimal designs have the minimum variance, their MSE may get too large due to the bias. In our study, we also consider a special class of designs called bias-optimal designs, which provide zero bias and hence, their MSE is completely based on the variance portion. Margolin (1969) demonstrates that folding over an efficient non-orthogonal resolution III design with n runs produces a non-orthogonal resolution IV design with $2n$ runs. Such a design has zero bias and provides the minimum variance among all non-orthogonal fold-over designs. For convenience, they are called best fold-over designs (BFDs) throughout our study. In folding over a design, all the factor levels are reversed to form runs that are mirror images of those in the initial factorial design.

Nevertheless, BFDs do exist up to $n/2$ factors for a design with n runs. More precisely, when the run size is 10, a BFD is available for up to five factors. We use the efficient non-orthogonal resolution III designs summarized by Margolin (1969) to produce the BFDs of run sizes 10, 12, 14, 18, 20 and 28 with the number of factors being 5, 6, 7, 9, 10 and 14 respectively. The BFD for $n = 10$ and $m = 5$ is constructed as below, where the second half of the design is a mirror image of the first half.

$$\begin{pmatrix} 1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 \\ \hdashline -1 & -1 & -1 & 1 & -1 \\ -1 & -1 & 1 & -1 & -1 \\ -1 & 1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 \end{pmatrix}$$

2.3 Nearly Orthogonal Arrays of Strength Three and Their Construction

In this section, we search for a set of alternative designs to use when OAs of strength three do not exist. Variance-optimal designs are efficient in optimizing the variance but their MSE may be too large due to the large bias. On the other hand, BFDs are bias-optimal, but the possible high variance may lead to a large MSE. We aim at constructing an alternative design, for each run size being considered in this study, to minimize both variance and bias to some extent. The design that has the lowest V_3 is chosen to be the best alternative design to use in our study. Such designs are similar to the OAs of strength three, as they are intended to minimize both bias and variance simultaneously in contrast to variance and bias-optimal designs. We call such factorial designs nearly orthogonal arrays (NOAs) of strength three.

We consider the construction of nearly orthogonal arrays of strength three under two scenarios. In the first scenario, we consider the situation of $m = n/2$, where both variance-optimal designs and BFDs are available. In the second scenario, we consider the number of factors in a design to be greater than the half of the number of runs by 1 ($m = n/2 + 1$), in which case BFDs are not available. The construction of NOAs of strength three is done separately under the two scenarios using two approaches, partially folding-over OAs of strength two and adding runs to OAs of strength two. Once the best design is found from the two approaches based on their V_3 values, a local search algorithm is used to improve it further. Those concepts are discussed in Subsections 2.3.1, 2.3.2 and 2.3.3, respectively.

2.3.1 Partial Fold-Over of OAs of Strength Two

In this approach, we start with OAs of strength two. For each run size, we identify all the OAs of strength two with smaller run sizes and the required number of factors and then partially fold over some of their runs to produce arrays with the required run size. Many of such arrays are generated by considering all possible permutations. The design which gives the smallest V_3 out of all the designs is considered the best under this approach. To illustrate, consider the run size 20 with 10 factors. There are two types of OAs of strength two with lower orders, 12 and 16 that contain the sufficient number of runs to fold over. We first start with an $\mathrm{OA}(12, 2^{10}, 2)$ and fold over eight runs out of its 12 runs in all possible ways. Then, the same steps are done on an $\mathrm{OA}(16, 2^{10}, 2)$ except the fact that here we fold over only four runs. Out of all the resulting arrays, we choose the design with the lowest V_3.

2.3.2 Adding Runs to OAs of Strength Two

Here, we choose OAs of strength two with lower orders and then add some runs to produce the required $n \times m$ design. To find an NOA with n runs and m factors, we begin with lower order OAs of strength two having n' runs and m factors, where $n' < n$. We then fill the remaining $n - n'$ runs with +1's and -1's by considering all the permutations under the constraint that the resulting design should be balanced in order to make the process less complicated. A balanced design is defined as a design, where +1's and -1's occur with the same frequency in every column. Thus, the J_1-component of such designs becomes zero, and hence, only J_2- and J_3-components contribute to the V_3. Consider the same example with 20 runs. We again begin with an $\mathrm{OA}(12, 2^{10}, 2)$ and an $\mathrm{OA}(16, 2^{10}, 2)$. Let the $\mathrm{OA}(12, 2^{10}, 2)$ be the first part of the resulting design. The second part is a matrix of 8×10, where we fill its columns with ten different vectors of four -1's and four +1's. The process is repeated by considering all the permutations. The same steps are done on the $\mathrm{OA}(16, 2^{10}, 2)$ to fill the remaining 4×10 empty matrix. Finally, the 20×10 array that gives the lowest V_3 out of all the resulting designs is considered to be the best under this approach.

2.3.3 Local Search Algorithm

The purpose of the local search algorithm is to improve the optimal design found from the two approaches mentioned above by comparing it with its one-unit away neighboring designs. We first choose the design with the smallest V_3 from the two approaches and call it

NOA_0. The one-unit away neighbors to NOA_0 are generated by swapping the signs of two elements with opposite signs in column i for all $i = 1, 2, ..., m$. For a design of order $n \times m$, there are $m(n/2)^2$ one-unit away neighboring designs in total. If the NOA_0 has the lowest V_3 among its neighbors, we stop the algorithmic search and use it as the best NOA in our study for n runs with m factors. If a neighboring design gives a V_3 smaller than that of NOA_0, we move to that particular design and call it NOA_1. Then the same steps are done on NOA_1. Similarly, the search continues until the V_3 of the current design is the lowest among its one-unit away neighbors. Finally, the design found to be the best from the local search algorithm is considered as the best NOA of strength three. The illustration below describes the first two stages of the iterative process.

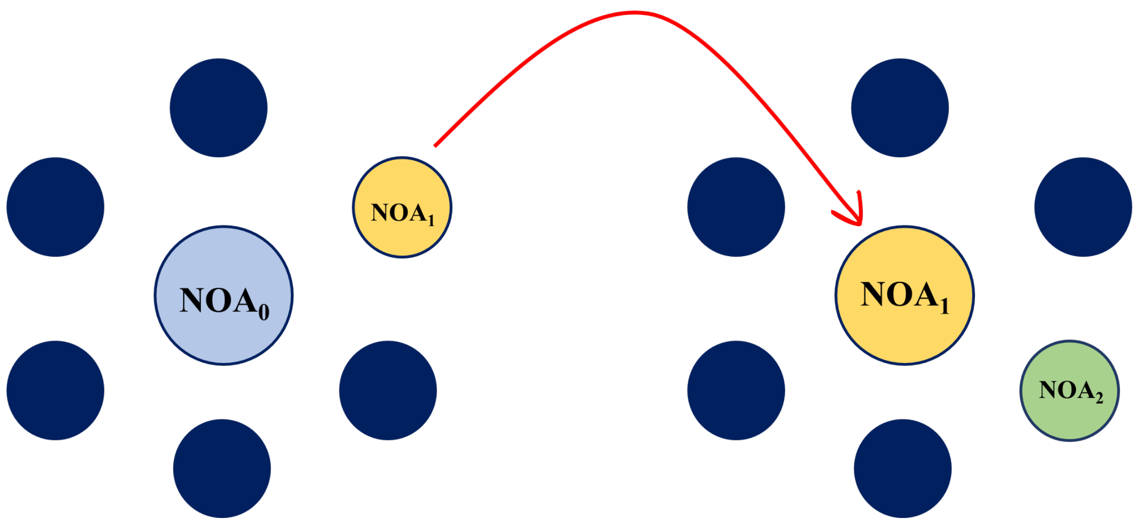

Figure 2.1: First Two Steps of the Local Search Algorithm

Consider the NOA_0 for 20 runs and 10 factors found by folding over four runs of an $\text{OA}(16, 2^{10}, 2)$. We then performed the local search algorithm to improve the NOA_0 by comparing it with its one unit away neighbors. The complete set of iterations is summarized below:

Design	NOA_0	NOA_1	NOA_2	NOA_3	NOA_4	Best NOA
V_3	50	48	46.5	46	45	45

Table 2.1: Complete Set of Iterations of the Local Search Algorithm For the 20×10 Experiment

2.4 An MSE Criterion for Comparing Different Designs

The three dimensional projection property (V_3) provides a goodness measure for a fractional factorial design. However, it is combinatorial. We need to carry out a direct statistical comparison of the candidate designs to provide a clear guideline for practitioners to use in order to choose one design over the others. We use the MSE of estimated main effects to compare the three classes of designs considered in our study. As we allow some two-factor interactions to be non-negligible in the model, the MSE should be computed under the linear model (1.3). BFDs are not affected by non-negligible two-factor interactions due to their zero bias. Therefore, only the variance contributes to the MSE. However, best NOAs and variance-optimal designs are affected by the unknown $\beta^{(2)}$.

Mukerjee and Tang (2012) adopted a Bayesian inspired approach to calculate the MSE by taking the effect sparsity into consideration. In factorial experiments, the principle of effect sparsity states that besides the main effects, only a few two-factor interactions are likely to be important. Under the two main assumptions that all the subsets of N two-factor interactions are equally likely to be significant and the active two-factor interactions are uncorrelated with mean zero and variance τ^2, we can express the expected MSE as below:

$$MSE = \sigma^2 tr(M) + \pi\tau^2 K_2, \tag{2.1}$$

where $\pi = \frac{N}{\binom{m}{2}}$ denotes the proportion of significant two-factor interactions, which is called the fraction of sparsity, $K_2 = tr(BB^T)$, and M and B matrices are previously defined in Section 1.3.

Chapter 3

Results of Comparisons

In this chapter, we summarize the important results of our study. We first describe how variance-optimal designs and bias-optimal designs perform using an example. The MSE is computed for both existing designs in 10 runs and 5 factors to illustrate their general behavior. In Section 3.1, we consider six experiments with 10, 12, 14, 18, 20 and 28 runs in $m = n/2$ factors. We conduct the same comparison for variance-optimal designs, bias-optimal designs and best NOAs of strength three in those experiments. The MSE calculation is done under seven different values of τ/σ, which is indicated by C throughout our study. For convenience, we take $\sigma = 1$ in all our calculations with $C = \tau/\sigma = \tau = 0.025$, 0.05, 0.1, 0.25, 0.5, 1 and 2. In every experiment, we also consider six different scenarios by changing the proportion of non-negligible two-factor interactions from 1 to 1/32. Thus, we take $\pi = 1$, 1/2, 1/4, 1/8, 1/16 and 1/32. Our goal here is to provide a range of C given by C_1 and C_2 such that the alternative design outperforms the corresponding two existing designs within the range of C_1 to C_2. In Section 3.2, we consider another six experiments having the same set of run sizes, but with $m = n/2 + 1$ factors, where the bias-optimal designs do not exist. Thus, we compare the available variance-optimal designs with the best NOAs for the same set of run sizes with 6, 7, 8, 10, 11 and 15 factors, respectively, by considering their MSEs. We provide a cut-off value C^* for C in each of the experiments so that the alternative design outperforms the corresponding variance-optimal design when $C > C^*$.

To understand the behaviour of variance-optimal designs and bias-optimal designs, we first examine in detail an example in which 5 factors are studied using 10 runs. We compute the MSE by taking effect sparsity into consideration as expressed in equation (2.1). Even though the variance of variance-optimal designs remains constant, the bias increases with the fraction of sparsity, which leads to the gradual increment in the MSE of estimated main effects. However, the MSE of bias-optimal designs remains unchanged as only the variance of such designs contributes to the MSE. This disproportionate behavior allows us to obtain a cut-off point, beyond which the best fold-over design performs better than the variance-optimal design of the same order. The example below summarizes the bias, variance and the MSE of estimated main effects of the two designs in an experiment with 10 runs and 5 factors when all two-factor interactions are significant.

n	m	$C = \tau/\sigma$	VOD			BFD		
			Variance	Bias	MSE	Variance	Bias	MSE
10	5	0.025	0.536	0.004	0.54	0.556	0	0.556
		0.05	0.536	0.014	0.55	0.556	0	0.556
		0.1	0.536	0.057	0.593	0.556	0	0.556
		0.25	0.536	0.353	0.889	0.556	0	0.556
		0.5	0.536	1.413	1.949	0.556	0	0.556
		1	0.536	5.653	6.189	0.556	0	0.556
		2	0.536	22.612	23.148	0.556	0	0.556

Table 3.1: Variance, Bias and MSE for Variance and Bias-Optimal Designs with 10 Runs and 5 Factors When All Two-Factor Interactions are Significant ($\pi = 1$)

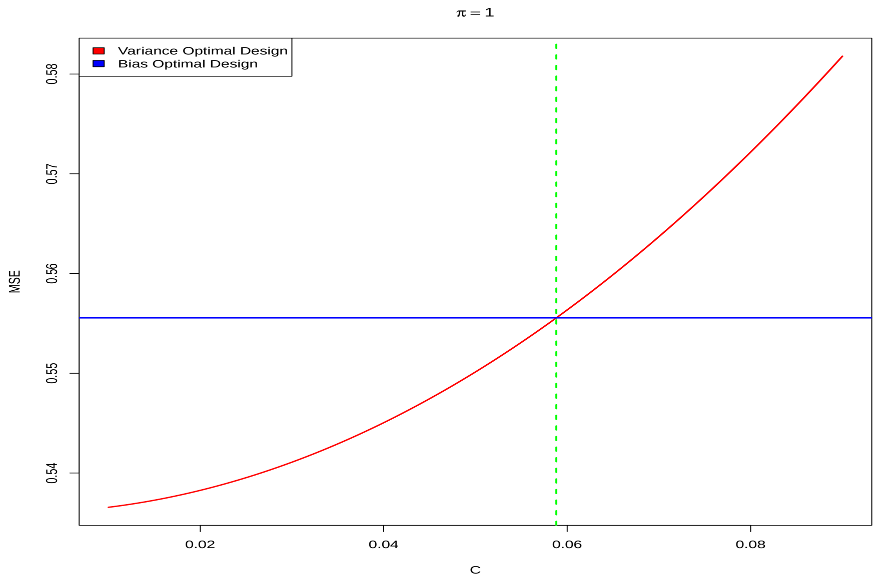

Figure 3.1: Comparison of the MSE for Variance and Bias-Optimal Designs with n=10 and m=5 when $\pi = 1$

Table 3.1 indicates that the MSE of the variance-optimal design is smaller at lower values of C, but gradually increases with τ to exceed the MSE of the bias-optimal design at a certain point, which is approximately 0.059 according to Figure 3.1. Hence, it is clear that the variance-optimal design is better at lower values of C, while the bias-optimal design outperforms the variance-optimal design when C exceeds 0.059. We can observe a similar trend for the two types of designs for other run sizes.

3.1 Comparison of Variance-Optimal Designs and Bias-Optimal Designs with Best Nearly Orthogonal Arrays of Strength Three

In this section, we carry out the same comparison by using the best NOAs of strength three. We choose designs with the lowest V_3 to be the best NOAs as discussed in Section 2.3. We consider six experiments having 10, 12, 14, 18, 20, 28 runs with 5, 6, 7, 9, 10, 14 factors respectively. In each experiment, we aim to compare all three types of designs together to identify three separate regions of C, in each of which one design performs better than the other two. The V_3 values and the corresponding J-components of the three types of designs in each experiment are displayed in Table 3.2. There are two best NOAs that perform equally well for run size 28. Hence, both of the designs are listed, as NOA 1 and NOA 2.

n	m	Design	V_3	J_1-Component	J_2-Component	J_3-Component
10	5	VOD	4.375	0.75	0.75	2.875
		BFD	1.875	0	1.875	0
		NOA	3.125	1.125	0.75	1.25
12	6	VOD	5	0	0	5
		BFD	6	0	6	0
		NOA	5	0	3	2
14	7	VOD	14.312	2.812	2.812	8.688
		BFD	59.063	0	59.063	0
		NOA	6.813	2.813	2.813	1.188
18	9	VOD	44.75	7	7	30.75
		BFD	29.75	0	29.75	0
		NOA	27.75	1.75	12.25	13.75
20	10	VOD	30	0	0	30
		BFD	50	0	50	0
		NOA	45	22.5	0	22.5
28	14	VOD	91	0	0	91
		BFD	126	0	126	0
		NOA 1	136.5	68.25	0	68.25
		NOA 2	136.5	63.375	9.75	63.375

Table 3.2: Three Dimensional Projection Property of Variance-Optimal Designs, Bias-Optimal Designs and Best NOAs in experiments with n=10, 12, 14, 18, 20, 28 where $m = n/2$

We use equation (1.1) to calculate all the values listed in Table 3.2. The orthogonal arrays of strength two are used as variance-optimal designs in experiments with 12, 20 and 28 runs, and hence their J_1- and J_2-components are zero. This is explainable as an OA of strength two is balanced and its columns are orthogonal to each other. Thus, only the J_3-component contributes towards the V_3. OAs of strength two are not available for run sizes 10, 14 and 18, where the corresponding variance-optimal designs are constructed using Lemma 1. In both situations, we can observe from Table 3.2 that the variance-optimal designs provide lower J_2-components. In contrast, J_3-components of bias-optimal designs are zero in every experiment, resulting in zero bias.

Nevertheless, the J_2-component of the best NOA in an experiment is lower than that of the bias-optimal design, indicating that it provides a variance lower than that of the bias-optimal design. Moreover, the best NOA provides a J_3-component lower than that of the variance-optimal design but considerably higher than that of the bias-optimal design. This contradictory relationship clearly explains the fact that the bias of a best NOA found in our study is lower than that of the corresponding variance-optimal design, while it is undoubtedly higher than the bias-optimal design. In conclusion, the above-explained bias-variance trade off among the set of three designs allows us to find three separate regions of C for each experiment. Let us use the experiment with five factors in ten runs to illustrate the mentioned behavior between the variance-optimal design, bias-optimal design and the best NOA of order 10×5 by changing the fraction of sparsity.

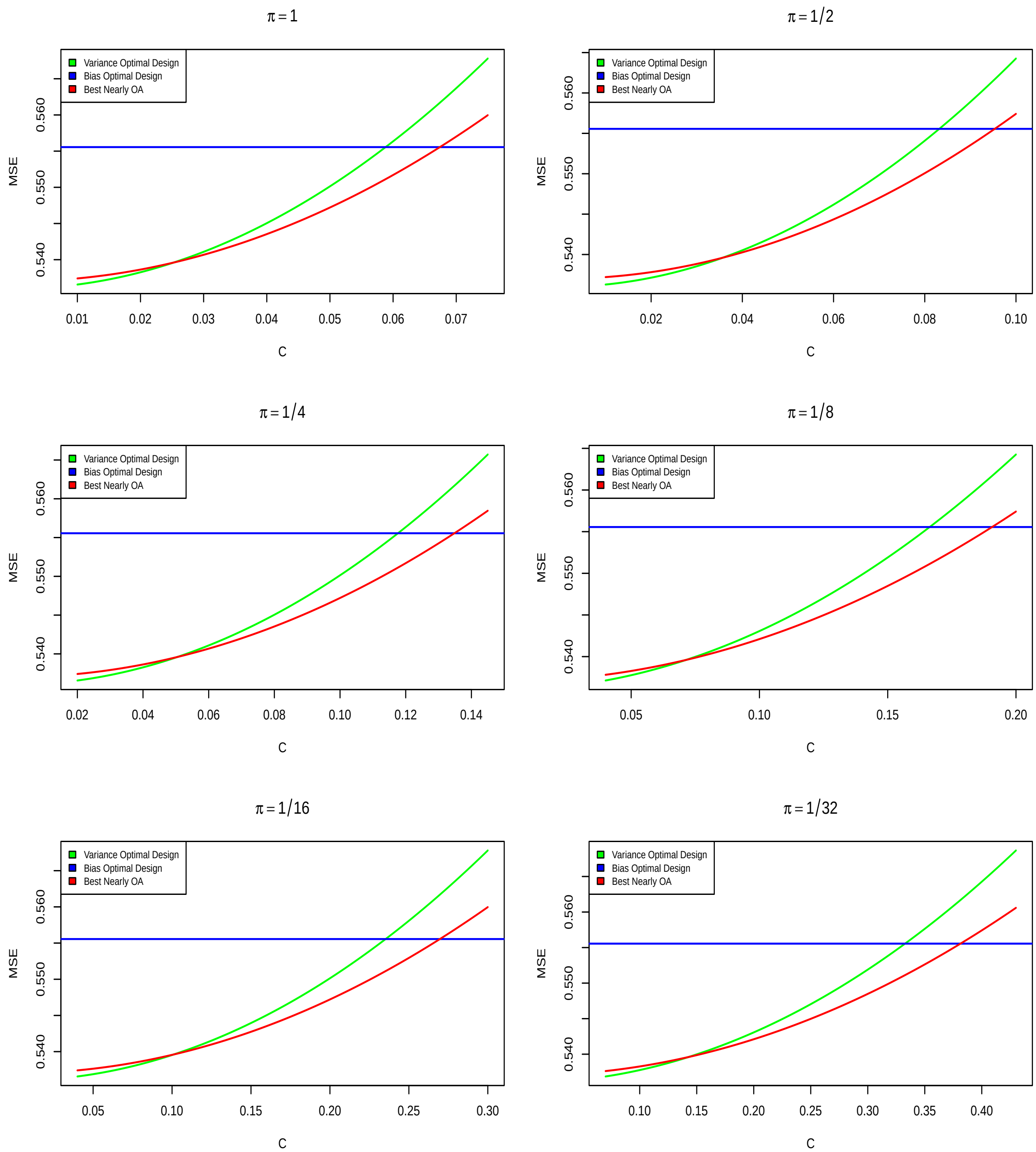

Figure 3.2: Comparison of the MSE for Designs With n=10 and m=5

According to Figure 3.2, the MSE of the variance-optimal design is the lowest for smaller values of C, which gradually increases to exceed the MSE of both the NOA and the bias-optimal design. Hence, the MSE of the variance-optimal design is the highest at higher values of C. The MSE of the best NOA slightly increases over the range of C and lies completely below the curve of the variance-optimal design after a certain point, while it eventually exceeds the constant MSE of the bias-optimal design. The MSE of both the variance-optimal design and the best NOA increases with the fraction of sparsity being considered in the study according to equation (2.1). For instance, when all the two-factor interactions are considered non-negligible, the two curves cross the curve of the best fold-over design at C values 0.059 and 0.067 respectively, whereas when only a half of the two-factor interactions are significant ($\pi = 1/2$), the intersections happen at slightly higher points, 0.083 and 0.095.

Figure 3.2 suggests three separate regions within each of which one out of the three designs performs better than the remaining two. Suppose the point where the curve of the variance-optimal design intersects that of the best NOA is indicated as C_1 and the intersecting point of curves of the best NOA and the bias-optimal design is named as C_2. Regardless of the fractions of sparsity being considered, the variance-optimal design and the best fold-over design perform the best when $C < C_1$ and $C > C_2$, respectively. The best NOA of the same order outperforms the two existing designs when $C_1 < C < C_2$. For example, the best NOA of order 10×5 outperforms the corresponding variance-optimal design and the bias-optimal design in the range of $0.025 < C < 0.068$ when none of the two-factor interactions are negligible, which means $\pi = 1$. Table 3.3 below summarizes the list of C_1 and C_2 values for all of the experiments listed in Table 3.2 for $\pi = 1, 1/2, 1/4, 1/8, 1/16, 1/32$. Figure 3.3 illustrates how the points C_1 and C_2 change when the fraction of sparsity increases. In conclusion, experimenters may use the listed regions to decide which design to use in their experiments.

- when $C < C_1$: Variance-optimal design is the best,
- when $C_1 < C < C_2$: Best NOA is the best,
- when $C > C_2$: Bias-optimal design (BFD) is the best.

n	m		$\pi = 1$	$\pi = 1/2$	$\pi = 1/4$	$\pi = 1/8$	$\pi = 1/16$	$\pi = 1/32$
10	5	C_1	0.025	0.036	0.051	0.071	0.101	0.143
		C_2	0.068	0.095	0.135	0.191	0.27	0.381
12	6	C_1	0.306	0.433	0.612	0.866	1.225	1.732
		C_2	0.339	0.479	0.677	0.957	1.354	1.915
14	7	C_1	0.064	0.091	0.128	0.182	0.257	0.363
		C_2	0.218	0.309	0.436	0.617	0.873	1.234
18	9	C_1	0.027	0.038	0.053	0.076	0.107	0.151
		C_2	0.111	0.158	0.223	0.315	0.446	0.63
20	10	C_1	0.048	0.068	0.095	0.135	0.19	0.269
		C_2	0.094	0.133	0.188	0.265	0.375	0.53
28	14	C_1	0.025	0.035	0.05	0.07	0.099	0.14
		C_2	0.045	0.064	0.09	0.128	0.181	0.255
		C_1	0.037	0.053	0.075	0.105	0.149	0.211
		C_2	0.043	0.061	0.087	0.122	0.173	0.245

Table 3.3: C_1 and C_2 at different values of π in experiments with $m = n/2$

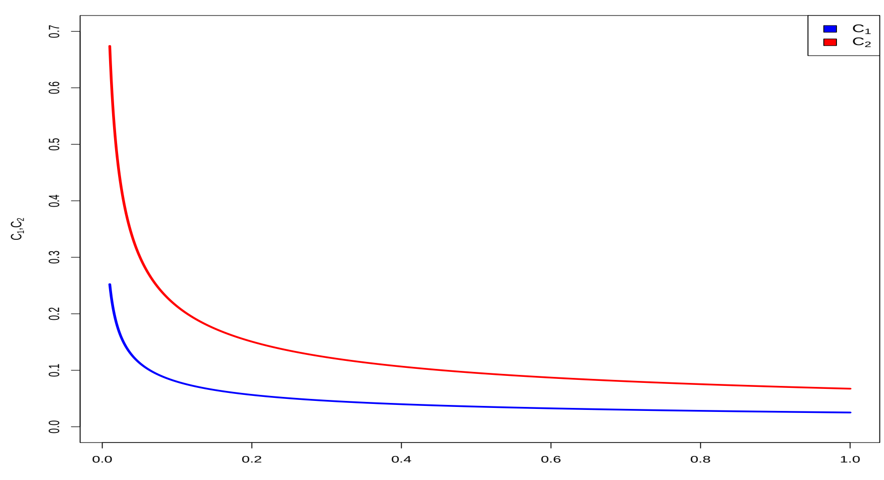

Figure 3.3: C_1 and C_2 of the experiment of 10×5 as a function of π

3.2 Comparison of Variance-Optimal Designs and Best Nearly Orthogonal Arrays of Strength Three When Best Fold-Over Designs Do Not Exist

Best fold-over designs do not exist when the number of factors in an experiment is greater than half the number of runs being considered. Therefore, only variance-optimal designs are available to use in such situations. Our goal in this section is to introduce some alternative designs for practitioners to use over variance-optimal designs. We provide a list of the best NOAs of strength three for experiments in $10, 12, 14, 18, 20, 28$ runs with $m = n/2 + 1$ factors. We use the same procedure explained in Section 2.3 to construct best NOAs. For each experiment, we aim to compare the variance-optimal design with the best NOA in terms of the MSE to provide two separate regions of C, in which each of the designs performs better than the other design. Table 3.4 contains the V_3 values and the J-components of the two designs in each experiment.

n	m	Design	V_3	J_1 Component	J_2 Component	J_3 Component
10	6	VOD	10	1.875	1.5	6.625
		NOA	7.75	0	3.75	4
12	7	VOD	8.75	0	0	8.75
		NOA	9.25	0	6.25	3
14	8	VOD	28	5.25	4.5	18.25
		NOA	20	2.625	6	11.375
18	10	VOD	64	11.25	10	42.75
		NOA	46.5	4.5	14.5	27.5
20	11	VOD	61.25	0	0	61.25
		NOA	69.75	0	33.75	36
28	15	VOD	171.75	0	0	171.75
		NOA	182.25	5.6875	11.375	165.1875

Table 3.4: Three Dimensional Projection Property of Variance-Optimal Designs and Best NOAs in Experiments Where $m = n/2 + 1$

According to Table 3.4, the J_2-component of the variance-optimal design is the lowest indicating the variance optimality. The best NOA contains the lowest J_3-component as it provides a smaller bias as compared to the variance-optimal design. To illustrate, we use the experiment of dimension 10×6 by changing $C = \tau/\sigma$ when $\pi = 1$.

n	m	$C = \frac{\tau}{\sigma}$	VOD			NOA		
			Variance	Bias	MSE	Variance	Bias	MSE
10	6	0.025	0.65	0.008	0.657	0.675	0.007	0.682
		0.05	0.65	0.031	0.680	0.675	0.026	0.701
		0.1	0.65	0.122	0.771	0.675	0.103	0.778
		0.25	0.65	0.762	1.411	0.675	0.645	1.32
		0.5	0.65	3.046	3.696	0.675	2.58	3.255
		1	0.65	12.184	12.833	0.675	10.32	10.995
		2	0.65	48.735	49.384	0.675	41.28	41.955

Table 3.5: Comparison of the MSE for the Variance-Optimal Design and the Best NOA With 10 Runs and 6 Factors When All Two-Factor Interactions are Significant

According to Table 3.5, the variance of estimated main effects of both designs remains unchanged, being unaffected by τ. We can clearly observe that the MSE of the variance-optimal design is smaller at lower values of C and it gradually increases to exceed the MSE of the best NOA when C increases. That suggests that best NOAs are better than variance-optimal designs at higher values of C and vise versa. For instance, between the two designs being compared in the above example of order 10×6, the best NOA performs better when $C > 0.117$. Thus, for each experiment, we can identify a cut-off point for C, which provides separate optimal regions for the two designs. If the boundary point is indicated by C^*, the best NOA is better if $C > C^*$ and the variance-optimal design performs well when $C < C^*$. At the point of C^*, the two designs perform equally well in terms of the MSE. Figure 3.4 illustrates the corresponding C^* values for the same experiment in different fractions of sparsity.

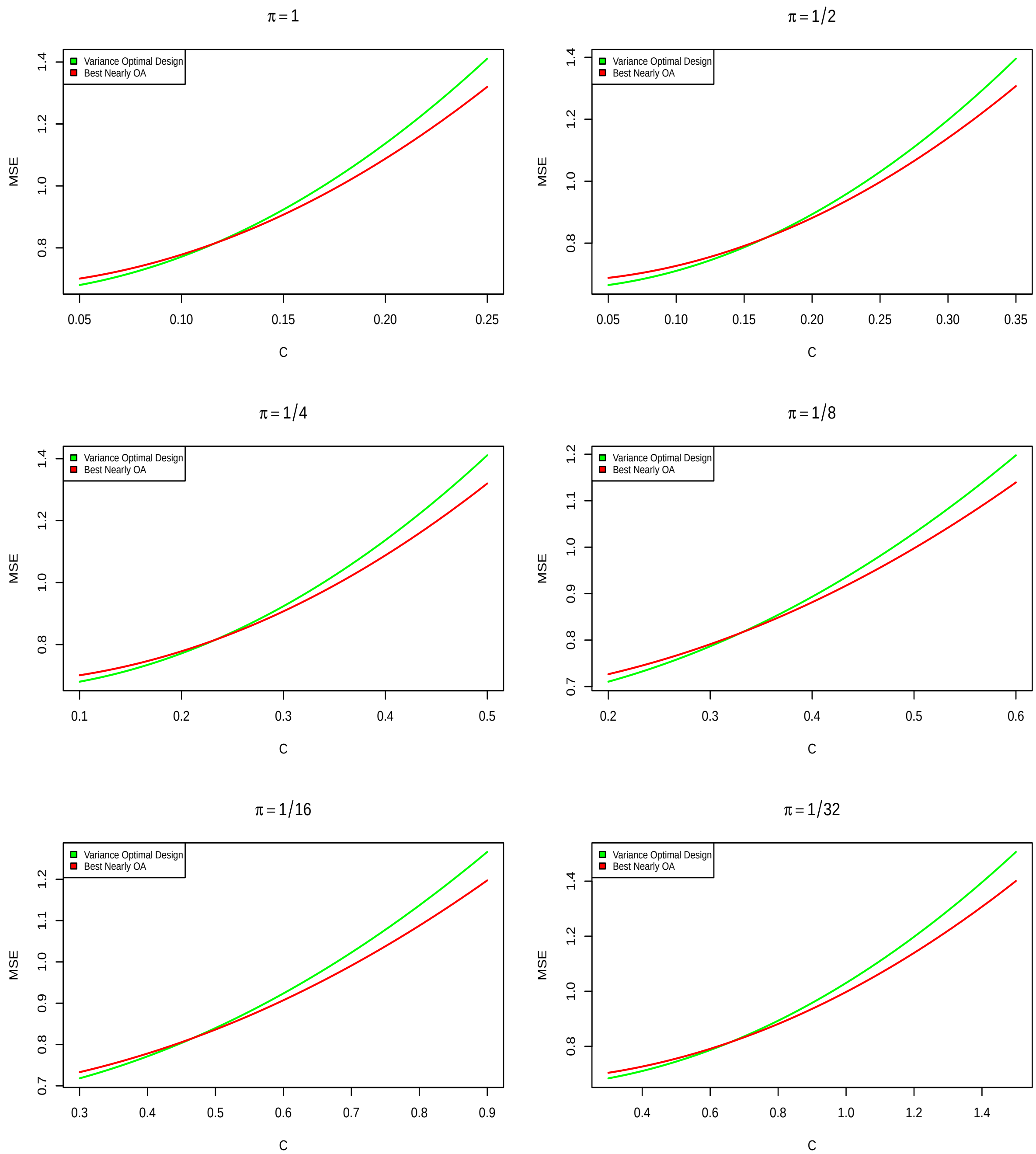

Figure 3.4: Comparison of the MSE of Designs with n=10 and m=6

As can be seen in Figure 3.5, the C^* value increases as π decreases. That is because the bias decreases when π decreases according to the equation (2.1), and hence the MSE of the estimated main effects is also getting decreased. Thus, the C value required at the intersection becomes higher. Table 3.6 below summarizes the list of C^* values for all experiments being considered in this section of our study with $\pi = 1, 1/2, 1/4, 1/8, 1/16, 1/32$. Experimenters may consider the corresponding regions to decide which design to use in practice in their experiments.

n	m	$\pi = 1$	$\pi = 1/2$	$\pi = 1/4$	$\pi = 1/8$	$\pi = 1/16$	$\pi = 1/32$
10	6	0.117	0.165	0.234	0.331	0.467	0.661
12	7	0.213	0.302	0.426	0.603	0.853	1.206
14	8	0.033	0.046	0.065	0.092	0.131	0.185
18	10	0.025	0.036	0.051	0.072	0.101	0.143
20	11	0.125	0.177	0.251	0.354	0.501	0.709
28	15	0.08	0.113	0.159	0.225	0.318	0.45

Table 3.6: C^* at different values of π in experiments with $m = n/2 + 1$

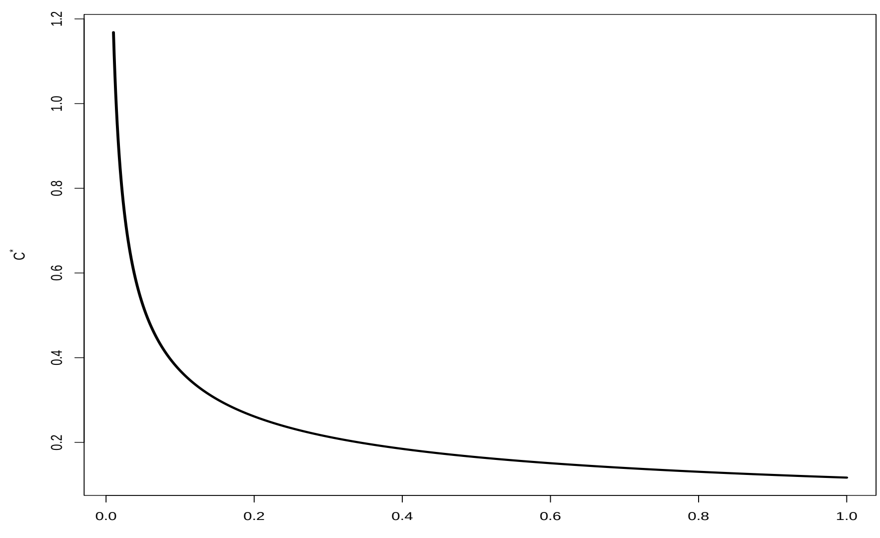

Figure 3.5: C^* of the experiment of 10×6 as a function of π

Chapter 4

Concluding Remarks

Variance-optimal designs and bias-optimal designs are available to use when OAs of strength three are not available. This study considers searching for an alternative class of designs called best NOAs of strength three that outperform the two existing classes of designs in some situations. In this study, we consider two scenarios; the number of factors being equal to half the number of runs and its being greater than half the number of runs by one. Construction of the best NOAs is done separately under those two scenarios using two approaches; partial folding-over of OAs of strength two and adding runs to OAs of strength two for run sizes 10, 12, 14, 18, 20 and 28. A local search algorithm is then used to improve the best design found from the two approaches. We conclude a design as a best NOA based on its three dimensional projection property. When the number of factors is greater than half the run size, BFDs are not available, and hence we compare the best NOAs with the available variance-optimal designs. Comparisons are done using the MSE of estimated main effects considering the effect sparsity. In conclusion, we provide a guideline for practitioners to use in practice to choose between different designs.

Even though we consider only up to 28 runs with 15 factors, the ideas of this study can easily be extended to larger designs.

Bibliography

Aloke Dey and Rahul Mukerjee. *Fractional factorial plans.* John Wiley & Sons, 1999.

Jacques Hadamard. Resolution d'une question relative aux determinants. *Bull. des sciences math.*, 2:240–246, 1893.

A Samad Hedayat, Neil James Alexander Sloane, and John Stufken. *Orthogonal arrays: theory and applications.* Springer Science & Business Media, 1999.

Barry H Margolin. Results on factorial designs of resolution iv for the 2^n and 2^n3^m series. *Technometrics*, 11(3):431–444, 1969.

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

Eric D Schoen, Nha Vo-Thanh, and Peter Goos. Two-level orthogonal screening designs with 24, 28, 32, and 36 runs. *Journal of the American Statistical Association*, 112(519): 1354–1369, 2017.

James Joseph Sylvester. Lx. thoughts on inverse orthogonal matrices, simultaneous signsuccessions, and tessellated pavements in two or more colours, with applications to newton's rule, ornamental tile-work, and the theory of numbers. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 34(232):461–475, 1867.

Boxin Tang. Theory of j-characteristics for fractional factorial designs and projection justification of minimum G_2-aberration. *Biometrika*, 88(2):401–407, 2001.

Boxin Tang and Lih-Yuan Deng. Minimum G_2-aberration for nonregular fractional factorial designs. *the Annals of Statistics*, 27(6):1914–1926, 1999.

Appendix A

List of Designs

A.1 Designs used in the case where $m = n/2$; the Variance-Optimal Design, Bias-Optimal Design and the Best NOA respectively.

For $n = 10$ **and** $m = 5$

$$
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 \\
-1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & -1 & -1
\end{pmatrix}
\qquad
\begin{pmatrix}
1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 \\
-1 & 1 & 1 & 1 & 1 \\
-1 & -1 & -1 & -1 & 1 \\
-1 & -1 & -1 & 1 & -1 \\
-1 & -1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 \\
1 & -1 & -1 & -1 & -1 \\
1 & 1 & 1 & 1 & -1
\end{pmatrix}
\qquad
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 \\
-1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & 1 & 1
\end{pmatrix}
$$

For $n = 12$ **and** $m = 6$

$$
\begin{pmatrix}
1 & -1 & -1 & -1 & 1 & -1 \\
1 & 1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & -1 \\
1 & -1 & 1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & -1 & -1 \\
-1 & 1 & 1 & -1 & 1 & 1
\end{pmatrix}
\qquad
\begin{pmatrix}
-1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & -1 & -1 & -1 & 1 \\
-1 & -1 & 1 & -1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & 1 \\
-1 & -1 & -1 & 1 & 1 & 1 \\
1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & 1 & 1 & 1 & -1 \\
1 & -1 & 1 & 1 & 1 & -1 \\
1 & 1 & -1 & 1 & 1 & -1 \\
1 & 1 & -1 & 1 & -1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1
\end{pmatrix}
\qquad
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 \\
1 & -1 & -1 & 1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 & -1 \\
1 & -1 & -1 & -1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & -1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & -1
\end{pmatrix}
$$

For $n = 14$ and $m = 7$

$$
\begin{pmatrix}
-1 & -1 & 1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & -1 & -1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 & 1 & -1 \\
-1 & 1 & -1 & 1 & 1 & -1 & 1 \\
1 & -1 & 1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 & -1 & -1 \\
-1 & 1 & 1 & -1 & 1 & 1 & 1 \\
-1 & 1 & 1 & 1 & -1 & -1 & -1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & -1 & -1 & -1 & -1
\end{pmatrix}
\quad
\begin{pmatrix}
1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & 1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & 1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & -1
\end{pmatrix}
\quad
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 & 1 \\
1 & 1 & 1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 & -1 & 1 \\
1 & -1 & -1 & -1 & -1 & 1 & 1 \\
-1 & -1 & 1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & 1 & -1 & 1 & -1
\end{pmatrix}
$$

For $n = 18$ and $m = 9$

$$
\begin{pmatrix}
-1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 \\
1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 \\
1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 \\
1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 \\
1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1
\end{pmatrix}
\begin{pmatrix}
1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\
1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 \\
-1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
-1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\
-1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 \\
1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\
1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\
-1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 \\
1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 \\
-1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1
\end{pmatrix}
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 \\
-1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1
\end{pmatrix}
$$

For $n = 20$ and $m = 10$

$$
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 \\
-1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\
-1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \\
1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1
\end{pmatrix}
\qquad
\begin{pmatrix}
1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\
-1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 \\
-1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 \\
-1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\
1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \\
1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 \\
1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\
-1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\
-1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -1
\end{pmatrix}
$$

$$
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 \\
-1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\
-1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \\
1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1
\end{pmatrix}
$$

For $n = 28$ **and** $m = 14$

$$
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\
-1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
-1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\
-1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\
-1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 \\
1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 \\
1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 \\
1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 \\
1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1
\end{pmatrix}
\begin{pmatrix}
1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\
-1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\
-1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\
-1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \\
1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\
-1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\
1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\
1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\
-1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 \\
-1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & 1 \\
1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1
\end{pmatrix}
$$

$$
\begin{pmatrix}
-1&-1&-1&-1&1&1&1&-1&-1&1&1&1&1&-1\\
-1&-1&1&1&1&-1&1&1&1&-1&1&-1&1&-1\\
-1&1&-1&-1&1&-1&-1&1&-1&-1&1&1&1&1\\
1&-1&-1&-1&-1&1&-1&1&1&-1&1&1&1&-1\\
1&1&1&-1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&1&-1&-1&-1&1&1&-1&-1&-1&1&1\\
1&-1&-1&1&-1&-1&1&1&-1&1&-1&1&1&-1\\
1&1&-1&1&1&-1&-1&-1&1&-1&1&1&-1&-1\\
-1&1&-1&-1&-1&-1&1&1&1&1&1&-1&-1&-1\\
1&1&-1&-1&-1&1&-1&-1&1&1&-1&1&-1&1\\
-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1\\
1&-1&-1&1&1&1&-1&1&-1&1&1&-1&-1&1\\
-1&1&-1&1&1&1&-1&1&1&1&-1&-1&1&-1\\
-1&1&1&1&-1&1&-1&1&-1&-1&1&1&-1&-1\\
1&1&-1&-1&1&1&1&1&-1&-1&-1&-1&-1&-1\\
1&-1&1&-1&1&-1&-1&1&1&1&-1&1&-1&-1\\
1&1&1&-1&-1&-1&-1&-1&-1&1&1&-1&1&-1\\
-1&1&1&-1&-1&1&1&-1&1&-1&-1&1&1&-1\\
1&-1&1&1&-1&1&1&-1&1&1&1&-1&-1&-1\\
1&-1&1&1&1&1&-1&-1&-1&-1&-1&1&1&-1\\
1&-1&-1&-1&1&-1&1&-1&1&-1&-1&-1&1&1\\
-1&-1&-1&1&-1&-1&-1&-1&1&1&1&1&1&1\\
-1&-1&-1&1&-1&1&1&1&1&-1&-1&1&-1&1\\
-1&-1&1&-1&1&1&-1&-1&1&-1&1&-1&-1&1\\
1&-1&1&-1&-1&-1&1&1&-1&-1&1&1&-1&1\\
-1&-1&1&-1&-1&1&-1&1&-1&1&-1&-1&1&1\\
-1&1&1&1&1&-1&1&-1&-1&1&-1&1&-1&1\\
1&1&-1&1&-1&1&1&-1&-1&-1&1&-1&1&1
\end{pmatrix}
\begin{pmatrix}
-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1&-1\\
1&1&-1&-1&1&1&-1&-1&1&1&1&-1&1&-1\\
1&-1&-1&1&-1&-1&1&-1&1&1&1&-1&1&1\\
-1&1&1&-1&-1&-1&1&1&1&-1&1&-1&1&-1\\
-1&-1&-1&-1&-1&1&1&-1&-1&1&1&1&1&-1\\
1&-1&1&-1&-1&1&-1&1&-1&1&-1&-1&1&1\\
1&1&-1&1&-1&1&1&1&-1&-1&-1&-1&-1&-1\\
-1&-1&1&1&-1&-1&1&1&-1&-1&1&1&-1&1\\
1&1&-1&-1&-1&1&-1&1&1&-1&1&1&-1&1\\
-1&-1&1&1&-1&1&-1&1&1&1&1&-1&-1&-1\\
1&1&1&1&-1&-1&-1&-1&-1&1&1&1&-1&-1\\
-1&1&-1&1&-1&-1&-1&1&1&1&-1&1&1&1\\
-1&1&1&1&1&-1&1&-1&-1&1&-1&-1&1&-1\\
-1&1&1&1&-1&1&-1&-1&-1&-1&1&-1&1&1\\
1&1&1&1&1&1&1&1&-1&1&1&1&1&1\\
-1&-1&-1&1&1&1&1&1&1&-1&-1&-1&1&1\\
1&-1&1&1&-1&1&-1&-1&1&-1&-1&1&1&-1\\
-1&1&1&-1&-1&1&1&-1&1&1&-1&-1&-1&1\\
1&1&-1&-1&-1&-1&1&-1&-1&-1&-1&1&1&1\\
1&-1&1&-1&1&1&1&-1&-1&-1&1&-1&-1&1\\
1&-1&-1&1&1&1&-1&-1&-1&1&-1&1&-1&1\\
1&1&1&1&1&-1&-1&-1&1&-1&-1&-1&-1&1\\
-1&-1&1&-1&1&-1&-1&-1&1&-1&1&1&1&1\\
-1&1&-1&-1&1&-1&-1&1&-1&1&1&-1&-1&1\\
-1&1&1&-1&1&1&-1&1&-1&-1&-1&1&1&-1\\
1&-1&-1&1&1&-1&-1&1&-1&-1&1&-1&1&-1\\
1&-1&1&-1&1&-1&1&1&1&1&-1&1&-1&-1\\
-1&1&-1&1&1&1&1&-1&1&-1&1&1&-1&-1
\end{pmatrix}
$$

A.2 Designs used in the case where $m = n/2+1$; the Variance-Optimal Design and the Best NOA respectively.

For $n = 10$ **and** $m = 6$

$$
\begin{pmatrix}
1&1&1&1&1&1\\
-1&1&-1&1&-1&1\\
-1&-1&1&1&-1&1\\
-1&-1&1&1&-1&-1\\
1&1&-1&-1&-1&1\\
-1&1&-1&-1&1&-1\\
-1&-1&-1&-1&1&1\\
-1&-1&1&-1&1&1\\
1&1&1&1&1&1\\
1&1&1&-1&-1&-1
\end{pmatrix}
\qquad
\begin{pmatrix}
1&1&1&1&1&1\\
-1&1&-1&1&-1&1\\
1&-1&-1&1&1&-1\\
-1&-1&1&1&-1&-1\\
1&1&1&-1&-1&-1\\
-1&1&-1&-1&1&-1\\
1&-1&-1&-1&-1&1\\
-1&-1&1&-1&1&1\\
-1&-1&-1&-1&-1&-1\\
1&1&1&1&1&1
\end{pmatrix}
$$

For $n = 12$ **and** $m = 7$

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ -1 & -1 & 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

For $n = 14$ **and** $m = 8$

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \end{pmatrix}$$

For $n = 18$ **and** $m = 10$

$$\begin{pmatrix} -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\ -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix} \qquad \begin{pmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 \\ -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

For $n = 20$ **and** $m = 11$

$$
\begin{pmatrix}
-1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 \\
-1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 \\
-1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 \\
1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 \\
1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 \\
-1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\
1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
-1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 \\
1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 \\
1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \\
1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1
\end{pmatrix}
\quad
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 \\
1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\
-1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 \\
1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 \\
-1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 \\
1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\
-1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1
\end{pmatrix}
$$

For $n = 28$ **and** $m = 15$

$$
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\
1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
-1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 \\
1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\
1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
-1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 \\
1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 \\
1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\
-1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\
1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\
-1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \\
1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 \\
-1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\
-1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 \\
-1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1
\end{pmatrix}
\begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\
-1 & -1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 \\
-1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \\
1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 \\
-1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 \\
1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 \\
-1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \\
1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \\
-1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 \\
1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\
1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 \\
1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 \\
-1 & 1 & -1 & 1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 \\
-1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\
-1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 \\
-1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \\
-1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 \\
1 & 1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 \\
-1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1
\end{pmatrix}
$$