DEPARTMENT OF ENGINEERING MANAGEMENT

Integer Programming Approaches to Find Row-Column Arrangements of Two-Level Orthogonal Experimental Designs

Nha Vo-Thanh, Peter Goos & Eric D. Schoen

# UNIVERSITY OF ANTWERP

## **Faculty of Applied Economics**



City Campus Prinsstraat 13, B.226 B-2000 Antwerp Tel. +32 (0)3 265 40 32 Fax +32 (0)3 265 47 99 www.uantwerpen.be

# **FACULTY OF APPLIED ECONOMICS**

DEPARTMENT OF ENGINEERING MANAGEMENT

## Integer Programming Approaches to Find Row-Column Arrangements of Two-Level Orthogonal Experimental Designs

Nha Vo-Thanh, Peter Goos & Eric D. Schoen

RESEARCH PAPER 2016-010 AUGUST 2016

University of Antwerp, City Campus, Prinsstraat 13, B-2000 Antwerp, Belgium Research Administration – room B.226 phone: (32) 3 265 40 32 fax: (32) 3 265 47 99 e-mail: joeri.nys@uantwerpen.be

The research papers from the Faculty of Applied Economics are also available at <u>www.repec.org</u> (Research Papers in Economics - RePEc)

## D/2016/1169/010

## Integer Programming Approaches to Find Row-Column Arrangements of Two-Level Orthogonal Experimental Designs

Nha Vo-Thanh

Faculty of Applied Economics, University of Antwerp, Belgium

Peter Goos

Faculty of Bioscience Engineering and Leuven Statistics Research Centre (LSTAT), KU Leuven, Belgium Faculty of Applied Economics and StatUa Center for Statistics, University of Antwerp, Belgium

Eric D. Schoen

Faculty of Applied Economics, University of Antwerp, Belgium TNO, Zeist, Netherlands

August 23, 2016

#### Abstract

Nonregular fractional factorial experimental designs offer flexibility in terms of run size as well as the possibility to estimate partially aliased effects. For this reason, there is much interest in finding good nonregular designs and in orthogonal blocking arrangements of these designs. In this contribution, we address the problem of finding orthogonal blocking arrangements in scenarios with two crossed blocking factors. We call these blocking arrangements of given two-level orthogonal treatment designs such that the treatment factors' main effects are orthogonal to both blocking factors. The first strategy involves a sequential approach which is especially useful when one blocking factor is more important than the other. The second strategy involves a simultaneous approach for situations where both blocking factors are equally important. For the latter approach, we propose three different optimization models, so that, in total, we consider four different methods to obtain row-column arrangements of the best two-level 24-run designs in terms of the *G*-aberration criterion. We compare the methods in terms of computing time and in terms of solution quality. We then apply the best approaches to 64- and 72-run orthogonal designs, and end the paper with a conclusion.

KEY WORDS: Aliasing; Confounding; Generalized Word-Length Pattern; Integer Linear Programming; Crossed Blocking Factors; Row-Column Design.

### 1 Introduction

Factorial experiments involve two or more treatment factors whose effects are of primary interest to the experimenter. They are often conducted under heterogeneous conditions. For instance, some experiments span multiple days or require different batches of material. In these cases, there is day-to-day variation or batch-to-batch variation. The technique recommended to deal with such sources of variation is blocking (Wu and Hamada, 2009) and the factor that defines the heterogeneous conditions is called a blocking factor. In the presence of one or more blocking factors, the experimental runs are grouped, each group is called a block and corresponds to a different level of a blocking factor. The goal of blocking is to minimize the dependence of the treatment factors' effect estimators on the differences between the blocks.

Blocking plays an important role in pharmaceutical, agricultural, food technology, and bioprocessing experiments. For example, the glasshouse experiment in Williams and John (1996) consists of six replicates of a design with two treatment factors along with a four-level and an eight-level blocking factor. The treatment factors were salt-irrigation level and seed lot. The two blocking factors were crossed and corresponded to the physical arrangement of the experimental units. The pastry dough experiment described by Gilmour and Trinca (2003) required seven days, and, within a day, the runs were performed at four different times. Therefore, the pastry dough experiment also involved two crossed blocking factors. Goos and Donev (2006a,b) mention a valve wear experiment, where the two crossed blocking factors are the valve position and the engine, and a food additives experiment, where the two crossed blocking factors are the enzyme supplier and the batch of wheat. Typically, experimental designs with two crossed blocking factors are called row-column designs, where the rows and the columns represent the levels of the first blocking factor and the levels of the second blocking factor, respectively.

Row-column designs for factorial experiments have been studied extensively. Jacroux and SahaRay (1990), for example, proposed a design construction method for two-level designs, which involve two settings for all the factors. They arranged either a full factorial design or a regular fractional factorial design involving m+n factors in  $2^m$  rows and  $2^n$  columns so that the resulting designs can be used to estimate all main effects independently of the block effects. Williams and John (1996) proposed a computer algorithm to construct row-column arrangements of factorial designs which allow a precise estimation of main effects, while ignoring the interaction effects.

Based on weighted mean efficiency factors, Gilmour and Trinca (2003) proposed an algorithm for creating factorial row-column designs for quantitative factors and a response surface model. Their construction involves an interchange algorithm (Jones and Eccleston, 1980), and treats the block effects (i.e., the row and column effects) as fixed. Goos and Donev (2006a,b) proposed a point-exchange algorithm to construct *D*-optimal row-column designs when the block effects are either treated as random or as fixed. The approaches of Gilmour and Trinca (2003) and Goos and Donev (2006a,b) are, however, not suitable for screening experiments with large numbers of factors and a limited number of runs, where the interest is in all main effects and all interaction effects. As a matter of fact, for these design approaches to be feasible, the number of experimental runs has to be large enough to estimate all effects of interest.

Based on the estimation capacity criterion, Cheng and Mukerjee (2003) provided a methodology to construct  $s^{n-k}$  fractional factorial designs in  $s^r$  rows and  $s^c$  columns. Their methods have two major limitations. First, the number of runs and the number of rows and columns must be powers of s. Second, because they use regular designs, the resulting row-column designs involve completely aliased effects.

In order to overcome these limitations, Vo-Thanh et al. (2016) proposed a general method to search for non-regular row-column designs with two-level treatment factors, starting from complete catalogs of non-isomorphic orthogonal arrays of the type  $OA(N, a \times b \times 2^n, t)$ , where N, a, b, n, and t represent the run size, the number of levels of the first blocking factor (corresponding to the rows of the row-column design), the number of levels of the second blocking factor (corresponding to the columns of the row-column design), the number of two-level treatment factors, and the strength of the design. An orthogonal array is of strength t if, for any given subset of t factors, all combinations of levels occur equally often. Unfortunately, for run sizes larger than or equal to 32, it is computationally infeasible to generate and explore complete catalogs of non-isomorphic orthogonal arrays of the type  $OA(N, a \times b \times 2^n, 2)$ . Therefore, searching for optimal row-column arrangements of orthogonal arrays of the type  $OA(N, 2^n, 2)$  using the methods of Vo-Thanh et al. (2016) is not feasible for  $N \ge 32$ . For this reason, for larger run sizes, Vo-Thanh et al. (2016) instead explore catalogs of strength-3 orthogonal arrays of the type  $OA(N, a \times b \times 2^n, 3)$ . However, the usefulness of this approach is limited because it requires both the full row-column design and the two-level treatment design to be of strength 3. This requirement is too strict because experimenters are generally not interested in interactions between the blocking factors and the treatment factors. Therefore, it is sufficient for the full row-column design to have strength 2, while the treatment design has strength 3. At present, no methodology is available for enumerating all possible ways in which all non-isomorphic strength-3 treatment designs of the type  $OA(N, 2^n, 3)$  can be embedded in a row-column design of the type  $OA(N, a \times b \times 2^n, 2)$ . Consequently, it is unknown how to optimally arrange, for example, the runs of the large number of attractive two-level treatment designs of strength 3 identified by Schoen and Mee (2012) in rows and columns to create row-column designs.

In the present paper, we cope with this void in the literature by taking a given orthogonal array of the type  $OA(N, 2^n, t)$  with good statistical features and arranging it in a rows and b columns such that the complete row-column design is an orthogonal array of the type  $OA(N, a \times b \times 2^n, 2)$ . To this end, we extend

the approach of Sartono et al. (2015, SSG), who propose a mixed integer linear programming approach for adding a single blocking factor to a two-level non-regular design. More specifically, we adapt their method to deal with two crossed blocking factors. Our main goal is to find row-column arrangements which allow the independent estimation of all main effects and block effects, and the estimation of as many twofactor interaction effects as possible. We consider two possible scenarios when searching for arrangements of two-level orthogonal designs in rows and columns. Both scenarios impose orthogonality between the main effects and the effects of the two blocking factors, corresponding to the rows and the columns. In Scenario 1, without loss of generality, we consider the rows to be more important than the columns, in the sense that the confounding of the interactions with the rows is more problematic than the confounding with the columns. Therefore, in Scenario 1, we minimize the confounding of two-factor interactions with the rows first, and, subject to this, we minimize the confounding of the interactions with the columns. In Scenario 2, we consider the rows and the columns to be equally important, in the sense that the confounding of the interactions with the rows is as problematic as the confounding with the columns.

The rest of this paper is structured as follows. In Section 2, we introduce the notation and the main concepts used. Next, in Section 3.1, we embed the mixed integer linear programming approach of SSG in a two-stage procedure for Scenario 1. More specifically, we arrange two-level treatment designs in rows first, and then in columns. We refer to this two-stage procedure as the sequential approach. In Section 3.2, we propose three different optimization models for Scenario 2, to arrange the two-level designs in rows and columns simultaneously. In Section 4, we apply the four approaches to construct 24-run row-column designs, compare the designs found with those from the literature, and study the computing times. This allows us to select the most appropriate approaches to search for row-column arrangements of 64- and 72-run treatment designs. In Section 5, we investigate whether using the row-column arrangement produced by the sequential approach as a starting solution for the simultaneous approach improves the computing time for the simultaneous approach. A discussion in Section 6 concludes the paper.

### 2 Preliminaries

In this section, we introduce the notation and the concepts used in the optimization models needed to arrange regular or nonregular two-level treatment designs in rows and columns. Following the literature on blocked experiments, we assume that there are no interactions between the first blocking factor (corresponding to the rows) and the second blocking factor (corresponding to the columns), and that there are no interactions between the blocking factors and the treatment factors. Therefore, only additive effects are required for the blocks (i.e., the rows and the columns) in the statistical model for the data obtained from the experiment. Also, since experimenters using two-level screening designs are generally only interested in main effects and two-factor interaction effects, we restrict our attention to these effects.

#### 2.1 Notation

We denote a given two-level factorial treatment design involving N runs and  $q_1$  factors by the  $N \times q_1$  matrix **X**. We denote the two levels of each treatment factor by -1 and +1. Every column of **X** is a main-effect contrast column. Every treatment design matrix **X** has a corresponding  $(N \times q_2)$ -dimensional two-factor interaction contrast matrix **W**, where  $q_2 = q_1(q_1 - 1)/2$  represents the number of two-factor interaction effects. The matrix **W** is obtained by element-wise multiplication of all pairs of main-effect contrast vectors in **X**. We denote the element in the *i*th row and *j*th column of **W** by  $w_{ij}$ .

Our primary goal is to arrange any given treatment design  $\mathbf{X}$  in rows and columns so that we can estimate a model containing all main effects, the block effects (corresponding to the rows and the columns), and as many two-factor interaction effects as possible. In addition, we desire the main-effect estimates to be independent from the block effects. To achieve these goals, (i) the main effects of the design  $\mathbf{X}$  have to be orthogonal to the blocks effects (i.e the row and the column effects), (ii) the row and the column blocking factors have to be orthogonal to each other, and (iii) the confounding between two-factor interaction effects and block effects needs to be minimized.

We denote the number of levels of the row factor by a, and the number of levels of the column factor by b. The assignment of the treatments to the rows is represented by the  $(N \times a)$ -dimensional binary matrix  $\mathbf{A}$ . An element  $a_{ij}$  of  $\mathbf{A}$  takes the value 1 when the *i*th run of the treatment design  $\mathbf{X}$  is assigned to the *j*th row,

and 0 otherwise. The assignment of the treatments to the columns is represented by the  $(N \times b)$ -dimensional matrix **B**. An element  $b_{ij}$  of **B** takes the value 1 when the *i*th run of **X** is assigned to the *j*th column, and 0 otherwise.

We quantify the confounding of the two-factor interaction effects with the rows using the  $(q_2 \times a)$ dimensional matrix  $\mathbf{S}_A = \mathbf{W}^T \mathbf{A}$  and the confounding of the two-factor interaction effects with the columns using the  $(q_2 \times b)$ -dimensional matrix  $\mathbf{S}_B = \mathbf{W}^T \mathbf{B}$ . We denote the elements of  $\mathbf{S}_A$  and  $\mathbf{S}_B$  by  $s_{ij}^A$  and  $s_{ij}^B$ , respectively. An element  $s_{ij}^A$  measures the extent to which the *i*th two-factor interaction is confounded with the *j*th level of the first blocking factor (i.e., with the *j*th row of the row-column arrangement). An element  $s_{ij}^B$  measures the extent to which the *i*th two-factor interaction is confounded with the *j*th level of the second blocking factor (i.e., with the *j*th column of the row-column arrangement). Ideally, all  $s_{ij}^A$  and  $s_{ij}^B$  values are zero, in which case there is no confounding of the treatments factors' second-order interactions with the blocks. The more positive or negative the  $s_{ij}^A$  and  $s_{ij}^B$  values, the more substantial is the confounding with the blocks (i.e., the rows and the columns). Finally, we denote the maximum absolute elements of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  by  $s_A$  and  $s_B$ , respectively.

#### 2.2 Quality measures for row-column two-level orthogonal designs

A row-column design includes two kinds of factors, treatment factors and blocking factors. We first review commonly used quality criteria for two-level treatment designs. Next, we discuss criteria for evaluating designs involving blocking factors.

#### 2.2.1 Two-level treatment designs

In this paper, we focus on orthogonal two-level treatment designs derived from orthogonal arrays. There may be many treatment designs with N runs, n factors and strength t. Designs that can be obtained from each other by row permutations, column permutations and sign switches in the columns are statistically equivalent and belong to the same isomorphism class. We denote a collection in which a single representative of every isomorphism class is included by  $OA(N, 2^n, t)$ , and we call this set the set of non-isomorphic designs with parameters N, n and t. Similarly, we denote by  $OA(N, a \times b \times 2^n, t)$  the set of non-isomorphic designs with N runs, n two-level factors, an extra factor with a levels, an extra factor with b levels and a strength of t.

For run sizes  $N \geq 32$ , it is computationally infeasible to generate and explore complete catalogs of non-isomorphic orthogonal arrays of the type  $OA(N, a \times b \times 2^n, 2)$  (see Schoen et al., 2010). Instead, we want to select good candidate treatment designs to be used as input to our methodology. One of the bestknown criteria for selecting a good two-level design is the *G*-aberration criterion originally proposed by Deng and Tang (1999). Using this criterion requires calculating the so-called  $J_k$ -characteristics of k-factor interaction contrast vectors, and involves the composition of frequency vectors describing the severity of the aliasing among main effects, two-factor interactions, three-factor interactions, etc. The minimum *G*aberration treatment design is the one that sequentially minimizes the entries of the frequency vectors, so that the least desirable kinds of aliasing are avoided as much as possible.

Tang and Deng (1999) proposed an alternative to the G-aberration criterion which they called the  $G_2$ aberration criterion. This criterion ranks designs based on the generalized word length pattern,  $W = (A_1, A_2, \ldots, A_n)$ . For treatment designs based on orthogonal arrays,  $A_1 = A_2 = 0$ . The entry  $A_3$  quantifies the extent to which main effects are aliased with two-factor interactions, the entry  $A_4$  quantifies the extent to which two-factor interactions are aliased with other two-factor interactions, etc. A minimum  $G_2$ -aberration design sequentially minimizes the generalized word length pattern. As experimenters' interest is usually only in main effects and two-factor interaction effects, it is common to consider only the  $A_3$  and  $A_4$  values when selecting an appropriate treatment design.

#### 2.2.2 Row-column designs

When a treatment design has to be blocked, the first concern is to avoid confounding of the main effects with the block effects. Therefore, the main effects have to be orthogonal to the blocks. The second concern is to ensure that the two-factor interaction effects are confounded as little as possible with blocks. One measure of the confounding of the treatment factors' interactions with the blocks is based on the mixed-type word count introduced in the literature by Cheng and Wu (2002) for finding blocking schemes in

scenarios involving a single blocking factor. The mixed-type word count was also used by Schoen et al. (2013) for problems involving one blocking factor only, and by Vo-Thanh et al. (2016) who identified rowcolumn arrangements of small treatment designs using a complete enumeration approach. When focusing on orthogonal blocking patterns, the most important mixed-type word counts quantify the confounding of the two-factor interactions with the first blocking factor (i.e., with the rows) and with the second blocking factor (i.e., with the columns). We denote these words counts by  $A_3^r$  and  $A_3^c$ . In this paper, we use the sum  $A_3^{r,c} = A_3^r + A_3^c$  as a measure of the total amount of confounding between the two-factor interactions and the blocks. Row-column designs with low  $A_3^{r,c}$  values are preferred, because, for these designs, there is little confounding between the interactions and the two blocking factors. Ideally, the  $A_3^{r,c}$  value is zero, in which case the two-factor interactions are orthogonal to both blocking factors.

One important issue with the  $A_3^r$  and  $A_3^c$  values is that they are hard to embed in a linear program: they are not linear functions of the decision variables  $a_{ij}$  and  $b_{ij}$ . In contrast, the elements of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  are linear functions of  $a_{ij}$  and  $b_{ij}$ . For this reason, in this paper, we quantify the confounding between the two-factor interactions, on the one hand, and the rows and columns, on the other hand, using the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$ . Row-column arrangements for which  $\mathbf{S}_A$  and  $\mathbf{S}_A$  are zero matrices have zero  $A_3^r$  and  $A_3^c$  values, and vice versa. Similarly, row-column arrangements for which the entries of  $\mathbf{S}_A$  and  $\mathbf{S}_A$  are small in absolute value generally have small  $A_3^r$  and  $A_3^c$  values. The methods we use to identify good row-column arrangements in this paper minimize the absolute values of all entries of  $\mathbf{S}_A$  and  $\mathbf{S}_A$ . More specifically, we first minimize the maximum absolute value of the elements of these matrices. Next, we minimize the sum of the absolute values of their elements. In doing so, we prioritize the avoidance of any severe confounding between the treatment factors' interactions and the blocks. Next, we try to minimize the remaining, less severe, confounding.

To illustrate the relationship between the  $A_3^r$  and  $A_3^c$  values, on the one hand, and the  $\mathbf{S}_A$  and  $\mathbf{S}_B$  matrices, on the other hand, consider a 24-run four-factor two-level design arranged in four rows and three columns. That arrangement is suitable if the first blocking factor has four levels, while the second blocking factor has three levels. In total, there exist 28,591 non-isomorphic orthogonal arrangements of four-factor two-level designs in four rows and three columns. This can be verified by enumerating all non-isomorphic arrays of the type OA(24, 4 × 3 × 2<sup>4</sup>, 2) using the algorithm in Schoen et al. (2010). Forty-five of these arrangements have an  $A_3^r$  value of 2/3 and an  $A_3^c$  value of zero. These 45 designs minimize the mixed-type word count  $A_3^r + A_3^c$  and are optimal in terms of this count. One of these designs is shown in Table 1. Because of the zero  $A_3^c$  value, the design allows the interaction effects to be estimated independently from the second blocking factors' effects, i.e., the column effects. The nonzero  $A_3^r$  value means that the interaction effects are confounded with the first blocking factors' effects, i.e., the row effects.

The  $\mathbf{S}_A$  and  $\mathbf{S}_B$  matrices of the row-column arrangement of the treatment design in Table 1 are shown in Table 2. The  $\mathbf{S}_B$  matrix is a zero matrix, confirming that the two-factor interactions are orthogonal to the second blocking factor. All entries of the  $\mathbf{S}_A$  matrix equal  $\pm 2$ , indicating that the two-factor interactions are partially confounded with the first blocking factor. The  $s_A$  and  $s_B$  values of the row-column arrangement under consideration are 2 and 0, respectively. The sum of all absolute  $s_{ij}^A$  values is 48, whereas that of all absolute  $s_{ij}^B$  values is zero.

## 3 Methodology

In this section, we propose two different strategies to search for row-column arrangements of a given two-level factorial treatment design  $\mathbf{X}$ . Regardless of the strategy, the row-column arrangements should possess the following characteristics:

- 1. The main effects of the treatment factors should be orthogonal to the rows.
- 2. The confounding of the two-factor interactions with the rows should be minimal.
- 3. The main effects of the treatment factors should be orthogonal to the columns.
- 4. The confounding of the two-factor interactions with the columns should be minimal.
- 5. The rows and columns should be orthogonal to each other.

Row	Column	$\mathbf{X}$	W		
0	0	1 -1 -1 1	-1 -1 1 1 -1 -1		
0	0	-1 $-1$ $-1$ $-1$	1  1  1  1  1  1		
0	1	1  1  1  -1	1  1  -1  1  -1  -1		
0	1	1  1  1  -1	1  1  -1  1  -1  -1		
0	2	-1 1 $-1$ $-1$	-1 1 1 $-1$ $-1$ 1		
0	2	1 -1 1 1	-1 1 1 $-1$ $-1$ 1		
1	0	-1 $-1$ $1$ $1$	1  -1  -1  -1  -1  1		
1	0	$1 \ -1 \ 1 \ -1$	-1 1 $-1$ $-1$ 1 $-1$		
1	1	1  1  -1  1	1  -1  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  -1  -1		
1	2	-1 $-1$ $-1$ $1$	1  1  -1  1  -1  -1		
2	0	-1 1 $-1$ 1	-1 1 $-1$ $-1$ 1 $-1$		
2	0	-1 1 1 $-1$	-1 $-1$ $1$ $1$ $-1$ $-1$		
2	1	1  -1  -1  -1	-1 $-1$ $-1$ $1$ $1$ $1$		
2	1	1 -1 1 1	-1 1 1 $-1$ $-1$ 1		
2	2	1  1  -1  1	1  -1  1  -1  1  -1		
2	2	-1 $-1$ $1$ $-1$	1  -1  1  -1  1  -1		
3	0	1  1  -1  -1	1  -1  -1  -1  -1  1		
3	0	1 $1$ $1$ $1$	1  1  1  1  1  1		
3	1	-1 $-1$ $-1$ $1$	1  1  -1  1  -1  -1		
3	1	-1 $-1$ $1$ $-1$	1  -1  1  -1  1  -1		
3	2	-1 1 1 1	-1 $-1$ $-1$ $1$ $1$ $1$		
3	2	1 -1 -1 -1	-1 -1 -1 1 1 1		

Table 1: A four-factor two-level treatment design with 24 runs arranged in four rows and three columns.

Table 2: Confounding matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  of the two-factor interaction contrasts of the design  $\mathbf{X}$  in Table 1 with the rows and columns.

	$\mathbf{S}$	A			$\mathbf{S}_B$	
-2	2	-2	2	0	0	0
2	2	-2	-2	0	0	0
2	-2	2	-2	0	0	0
2	-2	-2	2	0	0	0
-2	-2	2	2	0	0	0
2	-2	-2	2	0	0	0

The strategies extend the mixed integer linear programming approach of SSG in two different ways. The first strategy is a sequential approach suitable when the confounding of the two-factor interactions with one of the two blocking factors is of a greater concern than the confounding with the other blocking factor. Without loss of generality, we assume that the blocking factor corresponding to the rows of the row-column design is the factor of greater concern. The second strategy is a simultaneous approach to find row-column arrangements when the confounding with both blocking factors is of equal concern.

#### 3.1 Sequential approach

#### 3.1.1 Step 1

In the first step of the sequential approach, we arrange the given two-level treatment design  $\mathbf{X}$  in *a* rows using the mixed integer linear programming approach of SSG, which was intended for finding orthogonal blocking arrangements for problems involving one blocking factor. The resulting arrangement satisfies the goals (1) and (2).

To minimize the aliasing of the two-factor interactions with the rows, the approach of SSG sequentially minimizes the maximum absolute value of all elements of  $\mathbf{S}_A$ ,

$$s_A = \max\{|s_{ik}^A|, i = 1, \dots, q_2, k = 1, \dots, a\},\$$

and the sum of the absolute values of all elements of  $\mathbf{S}_A$ ,

$$\gamma = \sum_{i=1}^{q_2} \sum_{k=1}^a \left| s_{ik}^A \right|$$

The sequential minimization of  $s_A$  and  $\gamma$  is intended to ensure that no two-factor interaction is strongly confounded with the rows, and that, subsequently, all remaining confounding between the two-factor interactions and the rows is minimized.

Since linear programming approaches cannot deal with absolute values, we define the auxiliary nonnegative variables  $s_{ik}^{A+}$  and  $s_{ik}^{A-}$  for each element  $s_{ik}^{A}$  of  $\mathbf{S}_{A}$ . The variable  $s_{ik}^{A+}$  equals  $s_{ik}^{A}$  if  $s_{ik}^{A}$  is positive, and zero otherwise. The variable  $s_{ik}^{A-}$  equals  $-s_{ik}^{A}$  if  $s_{ik}^{A}$  is negative, and zero otherwise. Therefore, the variables  $s_{ik}^{A+}$  and  $s_{ik}^{A-}$  satisfy the equality

$$s_{ik}^A = s_{ik}^{A+} - s_{ik}^{A-}.$$

When expressed in terms of  $s_{ik}^{A+}$  and  $s_{ik}^{A-}$ , the secondary objective,  $\gamma$ , is

$$\gamma = \sum_{i=1}^{q_2} \sum_{k=1}^{a} (s_{ik}^{A+} + s_{ik}^{A-}).$$

Sequentially minimizing two objectives is known as pre-emptive goal programming or lexicographic goal programming in operations research. We implement our sequential minimization using the big-M method, where our primary objective,  $s_A$ , receives a large weight, M, and our secondary objective,  $\gamma$ , receives a weight of one. This leads to the following linear optimization model for Step 1 of our sequential approach:

$$\min f = Ms_A + \gamma = Ms_A + \sum_{i=1}^{q_2} \sum_{k=1}^{a} (s_{ik}^{A+} + s_{ik}^{A-}), \tag{1}$$

subject to

$$\sum_{i=1}^{N} w_{ji} a_{jk} - s_{ik}^{A+} + s_{ik}^{A-} = 0, i = 1, \dots, q_2; k = 1, \dots, a,$$
(2)

$$0 \le s_{ik}^{A+} \le s_A, i = 1, \dots, q_2; k = 1, \dots, a,$$
(3)

$$0 \le s_{ik}^{A-} \le s_A, i = 1, \dots, q_2; k = 1, \dots, a,$$
(4)

$$\mathbf{X}^T \mathbf{A} = \mathbf{0}_{q_1 \times a},\tag{5}$$

$$\mathbf{1}_{N}^{T}\mathbf{A} = (N/a)\mathbf{1}_{a}^{T},\tag{6}$$

$$\mathbf{A1}_a = \mathbf{1}_N,\tag{7}$$

$$a_{ij} \in \{0, 1\}, i = 1, \dots, N; j = 1, \dots, a.$$
 (8)

In the linear optimization model, Constraints (2)–(4) define the variables  $s_{ik}^{A+}$ ,  $s_{ik}^{A-}$  and  $s_A$  required to calculate the weighted objective function in Equation (1), using linear expressions. Constraint (5) states that only orthogonal blocking arrangements are allowed, i.e., blocking arrangements in which the treatment factors' main effects are orthogonal to the rows. Constraint (6) ensures that each of the *a* columns of **A** contains N/a treatments (in other words, that each row of the row-column arrangement has N/a runs), and Constraint (7) ensures that every treatment is assigned to exactly one row. In these constraints, the vectors  $\mathbf{1}_N$  and  $\mathbf{1}_a$  are vectors of ones of dimension N and a, respectively. Finally, the binary nature of the matrix **A**, which assigns the treatments to the rows, is imposed by Constraint (8), in which  $a_{ij}$  represents the element in the *i*th row and *j*th column of **A**.

All decision variables in this linear optimization model are integer for two-level treatment designs, so that the model belongs to the class of integer linear programming models. The first goal we try to achieve, goal (1), is enforced by Constraint (5). Goal (2), is achieved by minimizing the linear program's objective function in Equation (1).

#### 3.1.2 Step 2

Step 2 of the sequential approach starts from the optimal arrangement of the treatment design  $\mathbf{X}$  in *a* rows produced by Step 1 and arranges the treatment design  $\mathbf{X}$  in *b* columns as well, while leaving the arrangement in rows unchanged. Apart from one additional constraint, the linear optimization model in Step 2 has exactly the same structure as that in Step 1. The additional constraint in Step 2 ensures that the two blocking factors (i.e., the row factor and the column factor) are orthogonal to each other. This is necessary for the blocking factors to be crossed.

Denoting the elements of the confounding matrix  $\mathbf{S}_B$  between the treatment design  $\mathbf{X}$  and the columns by  $s_{ik}^B$ , the maximum absolute value of all elements by  $s_B$ , and the auxiliary variables corresponding to each element  $s_{ik}^B$  by  $s_{ik}^{B+}$  and  $s_{ik}^{B-}$ , the optimization model needed in Step 2 of the sequential approach is as follows:

$$\min f = Ms_B + \sum_{i=1}^{q_2} \sum_{k=1}^{b} (s_{ik}^{B+} + s_{ik}^{B-}),$$
(9)

subject to

$$\sum_{j=1}^{N} w_{ji} b_{jk} - s_{ik}^{B+} + s_{ik}^{B-} = 0, i = 1, \dots, q_2; k = 1, \dots, b,$$
(10)

$$0 \le s_{ik}^{B+} \le s_B, i = 1, \dots, q_2; k = 1, \dots, b,$$
(11)

$$0 \le s_{ik}^{B-} \le s_B, i = 1, \dots, q_2; k = 1, \dots, b,$$
(12)

$$\mathbf{X}^T \mathbf{B} = \mathbf{0}_{q_1 \times b},\tag{13}$$

$$\mathbf{A}_{\text{opt}}^T \mathbf{B} = (N/ab) \mathbf{J}_{a \times b},\tag{14}$$

$$\mathbf{1}_{N}^{T}\mathbf{B} = (N/b)\mathbf{1}_{b}^{T},\tag{15}$$

$$\mathbf{B1}_b = \mathbf{1}_N,\tag{16}$$

$$b_{ij} \in \{0,1\}, i = 1, \dots, N; j = 1, \dots, b,.$$
(17)

The linear optimization model in this step clearly also utilizes pre-emptive goal programming. This time, this is to prioritize the minimization of  $s_B$ . Constraints (10)–(12) define all the variables needed to calculate the objective function value in Equation (9). Constraint (13) ensures that the main effects of the treatment factors are all orthogonal to the columns. Constraint (14) is the new constraint which guarantees that the row blocking factor and the column blocking factor are orthogonal to each other. In that constraint, the matrix  $\mathbf{A}_{opt}$  corresponds to the optimal row arrangement identified in Step 1, and the matrix  $\mathbf{J}_{a\times b}$  represents the unit matrix of dimension  $a \times b$ . Constraints (15)–(17) ensure that the matrix **B** is binary, that every treatment is assigned to exactly one column of the design and that every column of the design contains N/b treatments. In Constraint (17),  $b_{ij}$  represents the element in row *i* and column *j* of the matrix **B**.

Note that, in Step 2 of the sequential approach, the matrix  $\mathbf{A}_{opt}$ , which indicates how the treatments are assigned to the *a* rows of the row-column design, is given. So, in the linear program defined by Equations (9)–(17), the elements of  $\mathbf{A}_{opt}$  are input parameters rather than decision variables, and all constraints are linear in the decision variables.

#### 3.2 Simultaneous approach

In this section, we propose three different optimization models to construct row-column designs when the confounding of the two-factor interactions with the two blocking factor is of equal concern. The first two of these models directly aim at optimizing the assignment matrices  $\mathbf{A}$  and  $\mathbf{B}$  for the given treatment design  $\mathbf{X}$ . This necessitates an additional constraint that enforces the orthogonality between the rows and the columns of the row-column arrangement. The first optimization model involves a quadratic constraint that ensures the orthogonality between the rows and the columns. The second optimization model involves a linearization of the quadratic constraint. The third optimization model avoids the extra orthogonality constraint by redefining the row-column arrangement problem as a permutation problem. In that model, the blocking structure is predefined, so that alternative row-column arrangements are represented as permutations of the treatment design  $\mathbf{X}$ . All three optimization models constitute a simultaneous approach that tackles the goals (1)–(5) at the same time.

#### 3.2.1 Simultaneous approach 1: A quadratic model

In this section, we describe our first optimization model to find row-column arrangements when the confounding with both blocking factors is of equal concern. The main goal is to find the two blocking matrices  $\mathbf{A}$  and  $\mathbf{B}$  that result in minimal confounding between the two-factor interaction effects, on the one hand, and the rows and the columns, on the other hand. A technical constraint when seeking the optimal  $\mathbf{A}$  and  $\mathbf{B}$  matrices is that these two matrices should be orthogonal to each other.

Achieving the goals (1), (3) and (5) is ensured by entering them as constraints in the optimization model. The goals (2) and (4) are dealt with in the optimization model's objective function. In the three models we present for the simultaneous approach, the objective function expresses our intention to make the two confounding matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  as small as possible simultaneously. In order to achieve this goal, we first minimize the maximum absolute value of all elements of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$ ,

$$s_{AB} = \max\{|s_{ij}^A|, |s_{ik}^B|, i = 1, \dots, q_2, j = 1, \dots, a, k = 1, \dots, b\}.$$

Next, we also minimize the sum of the absolute values of all elements of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$ ,

$$\gamma_{AB} = \sum_{i=1}^{q_2} \sum_{j=1}^{a} \left| s_{ij}^A \right| + \sum_{i=1}^{q_2} \sum_{k=1}^{b} \left| s_{ik}^B \right|.$$

As in the sequential approach in Section 3.1, we implement the sequential minimization of  $s_{AB}$  and  $\gamma_{AB}$ using pre-emptive goal programing. Also, we again use the auxiliary variables  $s_{ik}^{A+}$ ,  $s_{ik}^{A-}$ ,  $s_{ik}^{B+}$  and  $s_{ik}^{B-}$  to deal with the absolute values of the elements of  $\mathbf{S}_A$  and  $\mathbf{S}_B$ . We therefore calculate the sum of all of the absolute values of the elements of  $\mathbf{S}_A$  and  $\mathbf{S}_B$  as

$$\gamma_{AB} = \sum_{i=1}^{q_2} \sum_{k=1}^{a} (s_{ik}^{A+} + s_{ik}^{A-}) + \sum_{i=1}^{q_2} \sum_{k=1}^{b} (s_{ik}^{B+} + s_{ik}^{B-})$$

in our simultaneous optimization model:

$$\min f = Ms_{AB} + \gamma_{AB} = Ms_{AB} + \sum_{i=1}^{q_2} \sum_{k=1}^{a} (s_{ik}^{A+} + s_{ik}^{A-}) + \sum_{i=1}^{q_2} \sum_{k=1}^{b} (s_{ik}^{B+} + s_{ik}^{B-}),$$
(18)

subject to

$$\sum_{j=1}^{N} w_{ji} a_{jk} - s_{ik}^{A+} + s_{ik}^{A-} = 0, i = 1, \dots, q_2; k = 1, \dots, a,$$
(19)

$$\sum_{i=1}^{N} w_{ji} b_{jk} - s_{ik}^{B+} + s_{ik}^{B-} = 0, i = 1, \dots, q_2; k = 1, \dots, b,$$
(20)

$$0 \le s_{ik}^{A+} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, a,$$
(21)

$$0 \le s_{ik}^{A^-} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, a,$$
(22)

$$0 \le s_{ik}^{B+} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, b,$$
(23)

$$0 \le s_{ik}^{B^-} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, b,$$
(24)

$$\mathbf{X}^T \mathbf{A} = \mathbf{0}_{q_1 \times a},\tag{25}$$

$$\mathbf{X}^{T} \mathbf{B} = \mathbf{0}_{q_{1} \times b},$$

$$\mathbf{A}^{T} \mathbf{B} = (N/ab) \mathbf{J}_{a \times b},$$
(26)
(27)

$$\mathbf{1}_{N}^{T}\mathbf{A} = (N/a)\mathbf{1}_{a}^{T},$$
(28)

$$\mathbf{1}_{N}^{T}\mathbf{B} = (N/b)\mathbf{1}_{b}^{T},\tag{29}$$

$$\mathbf{A}\mathbf{I}_a = \mathbf{1}_N,\tag{30}$$
$$\mathbf{B}\mathbf{I}_L = \mathbf{1}_N$$

$$a_{ij} \in \{0, 1\}, i = 1, \dots, N; j = 1, \dots, a,$$
(32)

$$b_{ij} \in \{0,1\}, i = 1, \dots, N; j = 1, \dots, b.$$
(33)

Constraints (19)–(24) define all the variables needed to calculate the objective function value in Equation (18). Constraints (25) and (26) ensure that the main effects of the treatment factors are all orthogonal to the rows and the columns, respectively. Constraint (27) ensures that the row blocking factor and the column blocking factor are orthogonal to each other. Constraints (28) and (29) ensure that every row contains N/atreatments and that every column contains N/b treatments. Constraints (30) and (31) ensure that every treatment is assigned to exactly one row and to exactly one column, and Constraints (32) and (33) ensure that the matrices **A** and **B** are binary.

In the optimization model defined by Equations (18)–(33), Constraint (27) is quadratic, because it involves products of decision variables, namely the elements of the binary matrices **A** and **B**. This is in contrast with Constraint (14) in Step 2 of the sequential approach, where the matrix **A** was fixed after the sequential approach's Step 1 and the matrix **B** was the only one to be optimized.

#### 3.2.2 Simultaneous approach 2: The linearized quadratic model

Generally, solving optimization models involving quadratic constraints takes more computing time than solving models involving only linear constraints. It is, therefore, useful to try to replace quadratic constraints with linear ones.

In the case of Constraint (27), where the decision variables are binary, it is indeed possible to replace the quadratic expression with several linear constraints. To see this, note first that the original quadratic constraint is equivalent to

$$\sum_{k=1}^{N} a_{ki} b_{kj} = \frac{N}{ab}$$

for each row i and each column j of the row-column design. In the second simultaneous approach, we replace this quadratic expression by a new constraint,

$$\sum_{k=1}^{N} c_{ijk} = \frac{N}{ab},\tag{34}$$

Table 3: Creation of the new variable  $c_{iik}$  from  $a_{ki}$  and  $b_{ki}$  by using Inequalities (35)–(38).

			· · <b>j</b>
$a_{ki}$	$b_{kj}$	$a_{ki} + b_{kj} - 1$	$c_{ijk}$
0	0	-1	0
0	1	0	0
1	0	0	0
1	1	1	1

for each row i and each column j. The new constraint involves a new decision variable  $c_{ijk}$ , which we define as

$$c_{ijk} = a_{ki}b_{kj}$$

The new decision variable is a product of two binary variables  $a_{ki}$  and  $b_{kj}$ . Consequently, it should also be binary. More specifically,  $c_{ijk}$  should take the value 1 if both  $a_{ki}$  and  $b_{kj}$  are 1, and zero otherwise. Now, instead of calculating  $c_{ijk}$  as a product of  $a_{ki}$  and  $b_{kj}$ , it can also be determined by using the following set of linear inequality constraints:

$$c_{ijk} \le a_{ki}, i = 1, \dots, a; j = 1, \dots, b; k = 1, \dots, N,$$
(35)

$$c_{ijk} \le b_{kj}, i = 1, \dots, a; j = 1, \dots, b; k = 1, \dots, N,$$
(36)

$$c_{ijk} \ge a_{ki} + b_{kj} - 1, i = 1, \dots, a; j = 1, \dots, b; k = 1, \dots, N,$$
(37)

$$c_{ijk} \ge 0, i = 1, \dots, a; j = 1, \dots, b; k = 1, \dots, N.$$
(38)

Given that  $c_{ijk}$  is non-negative, Inequalities (35) and (36) ensure that  $c_{ijk}$  takes the value zero when  $a_{ki}$  or  $b_{kj}$  are zero. Inequality (37) ensures that  $c_{ijk}$  takes the value one when  $a_{ki}$  and  $b_{kj}$  are both one. The crucial role played by the sum  $a_{ki} + b_{kj} - 1$  in Inequality (37) is clarified in the final column of Table 3. That table shows all possible combinations of  $a_{ki}$  and  $b_{kj}$  values, as well the corresponding values of  $a_{ki} + b_{kj} - 1$  and  $c_{ijk}$ . The table allows us to verify that Constraints (35)–(38) do a good job at reproducing the value of the product  $c_{ijk} = a_{ki}b_{kj}$ , even though they do not involve products of  $a_{ki}$  and  $b_{kj}$  values.

It is clear that replacing the quadratic constraint in Equation (27) by Constraints (34)-(38) in the optimization model defined by Equations (18)-(33) results in a new optimization model that has the same optimal solution. A key difference between the new model and the original one is that the former only involves linear constraints, which is generally considered to be an advantage, despite the fact that the new formulation involves a larger number of constraints as well as a larger number of decision variables.

#### 3.2.3 Simultaneous approach 3: A permutation-based model

In this section, we reformulate the problem of finding an optimal row-column arrangement as a permutation problem. Rather than optimizing the binary blocking matrices **A** and **B** from scratch, while ensuring that the corresponding blocking structure (the pattern of the rows and the columns) is orthogonal, the permutation problem formulation considers the blocking structure as given. Finding an optimal row-column arrangement of a treatment design then comes down to assigning the N treatments to the N positions in the predefined blocking structure. Therefore, we have to find a permutation of all N treatments, where the treatment appearing first in the permutation is assigned to the first position in the blocking structure (first experimental run at the first level of the first blocking factor and the first level of the second blocking factor), the treatment appearing second is assigned to the second position, etc. The treatment appearing last in the permutation is assigned to the last position in the blocking structure (last experimental run at the *a*th level of the first blocking structure (last experimental run at the *b*th level of the second blocking factor).

Every permutation of a set of N objects can be represented by a permutation matrix **P**, i.e., an Ndimensional binary square matrix, involving exactly one entry of 1 in every row and in every column. For a given treatment design **X** with interaction contrast matrix **W** and a given orthogonal blocking structure [**A B**], the confounding matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  can be expressed in terms of the permutation matrices as well:  $\mathbf{S}_A = \mathbf{W}^T \mathbf{P} \mathbf{A}$  and  $\mathbf{S}_B = \mathbf{W}^T \mathbf{P} \mathbf{B}$ . Similarly, the conditions that the main effects of the treatment factors should be orthogonal to the rows and the columns of the row-column design can be reformulated as  $\mathbf{X}^T \mathbf{P} \mathbf{A} = \mathbf{0}_{q_1 \times a}$  and  $\mathbf{X}^T \mathbf{P} \mathbf{B} = \mathbf{0}_{q_1 \times b}$ , respectively. We denote the element in row *i* and column *j* of the permutation matrix  $\mathbf{P}$  by  $p_{ij}$ . This element takes the value 1 if the *i*th treatment in  $\mathbf{X}$  is assigned to the *j*th position in the blocking structure. The integer linear programming model based on the permutation matrix is then as follows:

$$\min f = Ms_{AB} + \gamma_{AB} = Ms_{AB} + \sum_{i=1}^{q_2} \sum_{k=1}^{a} (s_{ik}^{A+} + s_{ik}^{A-}) + \sum_{i=1}^{q_2} \sum_{k=1}^{b} (s_{ik}^{B+} + s_{ik}^{B-}),$$
(39)

subject to

$$\sum_{j=1}^{N} \left( \sum_{m=1}^{N} w_{mi} p_{mj} \right) a_{jk} - s_{ik}^{A+} + s_{ik}^{A-} = 0, i = 1, \dots, q_2; k = 1, \dots, a,$$

$$\tag{40}$$

$$\sum_{j=1}^{N} \left( \sum_{m=1}^{N} w_{mi} p_{mj} \right) b_{jk} - s_{ik}^{B+} + s_{ik}^{B-} = 0, i = 1, \dots, q_2; k = 1, \dots, b,$$
(41)

$$0 \le s_{ik}^{A+} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, a,$$
(42)

$$0 \le s_{ik}^{A-} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, a,$$
(43)

$$0 \le s_{ik}^{B+} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, b,$$
(44)

$$0 \le s_{ik}^{B^-} \le s_{AB}, i = 1, \dots, q_2; k = 1, \dots, b,$$
(45)

$$\mathbf{X}^T \mathbf{P} \mathbf{A} = \mathbf{0}_{q_1 \times a},\tag{46}$$

$$\mathbf{X}^T \mathbf{P} \mathbf{B} = \mathbf{0}_{q_1 \times b},\tag{47}$$

$$\mathbf{P}\mathbf{1}_N = \mathbf{1}_N, \tag{48}$$
$$\mathbf{1}_N^T \mathbf{P} = \mathbf{1}_N^T, \tag{49}$$

$$p_{ij} \in \{0, 1\}, i, j = 1, \dots, N.$$
 (50)

The objective function in the new formulation is exactly the same as that in the two other models for the simultaneous approach in Sections 3.2.1 and 3.2.2. Constraints (40)–(45) in the new formulation define the various components of the objective function. The differences between these constraints and those in the previous formulations is that Constraints (40) and (41) now make use of the elements of the permutation matrix, and that **A** and **B** are now given matrices defining two blocking factors that are orthogonal to each other rather than matrices with decision variables. More specifically,  $\mathbf{A} = \mathbf{I}_a \otimes \mathbf{1}_{N/a}$  and  $\mathbf{B} = \mathbf{1}_a \otimes \mathbf{I}_{N/(ab)}$ , where  $\mathbf{I}_a$  and  $\mathbf{I}_b$  are identity matrices of dimension a and b, respectively. Constraints (46) and (47) ensure that the treatment factors' main effects are orthogonal to the rows and columns, respectively. Finally, constraints (48)–(50) define the technical properties of the permutation matrix **P**. The final constraint enforces **P** to be a binary matrix, and Constraints (48) and (49) ensure that there is a one in every row and in every column of the permutation matrix.

A key difference between the optimization model defined by Equations (39)–(50) and the models in Sections 3.2.1 and 3.2.2 is that the elements of the matrices **A** and **B** are given, so that the permutation-based approach does not involve quadratic constraints.

### 4 Computational results

In this section, we apply the four optimization models, one for the sequential approach from Section 3.1 and three for the simultaneous approach from Section 3.2, to three different kinds of problem. First, we study the arrangement of 24-run orthogonal two-level treatments designs in four rows and three columns. Next, we study arrangements of 64-run orthogonal two-level designs in four rows and three columns. Finally, we study arrangements of 72-run orthogonal two-level designs in three rows and three columns. We selected these three specific kinds of problem because benchmark results are available from the literature. In particular, Vo-Thanh et al. (2016) used a complete enumeration to identify good row-column arrangements of strength-2 24-run two-level designs and strength-3 64- and 72-run two-level designs. The approach presented here is much more generally applicable than the complete enumeration approach of Vo-Thanh et al. (2016), but we believe that demonstrating that the sequential and simultaneous optimization approaches produce the same quality of designs as the complete enumeration approach will convince the reader of their power.

In all our comparisons, we pay attention to the quality of the row-column arrangements produced as well as the required computing time. For all of the cases studied, we set the value of M in the pre-emptive goal programming to 10,000. We performed all computations under Windows 7 (64-bit) on a Intel core i7-3770 PC with a 3.4GHz CPU for the 24-run designs and with a 2.6GHz CPU for the other designs, and an internal memory of 16 GB, using MATLAB 2012b along with CPLEX version 12.6.1. We used a time limit of 10,000 seconds for each approach for each problem tackled.

Following Vo-Thanh et al. (2016), we assume in the comparisons we make in this section that the confounding of two-factor interactions with the row factor and with the column factor is of equal concern to the experimenter. Therefore, our evaluation of the different optimization approaches is based on the objective function in Equation (18) (which is identical to that in Equation (39)). It should be pointed out, however, that Vo-Thanh et al. (2016) used different objective functions, based on the mixed-type words counts mentioned in Section 2.2.2.

#### 4.1 24-run designs

For our computational experiments involving 24-run two-level designs, we use the two-level treatment designs from the  $W_2$ - and  $W_3$ -optimal row-column arrangements identified by Vo-Thanh et al. (2016). This is because these designs perform very well in terms of the G- or  $G_2$ -aberration criterion, and because we can use the row-column arrangements of Vo-Thanh et al. (2016) as benchmarks. In total, we consider 17 different twolevel treatment designs. The smallest design involves four factors, while the two largest designs involve 13 factors. The four-factor design is the one shown in Table 1.

The objective function values of the row-column arrangements of the 24-run designs obtained by the four optimization approaches from Section 3 are listed in Table 4. The table's first column identifies the designs arranged in rows and columns. The designs' IDs are of the form  $q_1 i$ , where  $q_1$  denotes the number of two-level treatment factors and i is a label distinguishing designs with the same numbers of factors. The next three columns show the objective function values produced by the sequential approach, the computing time that approach required, and the  $A_3^{r,c}$  value quantifying the confounding between the treatment factors' interactions and both blocking factors. The next column shows the objective function values produced by the quadratic programming model, the linearized quadratic programming model and the permutation-based model, and the  $A_3^{r,c}$  value of the corresponding row-column arrangement. Remarkably, each of the three models for the simultaneous approach resulted in the same value for the objective function and the same  $A_3^{r,c}$  value except for treatment design 10.2. Coincidentally, the  $A_3^{r,c}$  value for the row-column arrangement of treatment design 10.2 obtained from the quadratic model and from the permutation-based model is 16.17, while the  $A_3^{r,c}$  value obtained from the linearized quadratic model is 16.61. All of the objective function values are optimal, because the linearized quadratic model and the permutation-based model were both solved to optimality well within the computing time limit of 10,000 seconds for each treatment design studied. The quadratic model returned the same objective function values as the other models. In nine of the 17 cases tackled with the quadratic model approach, CPLEX confirmed the optimality of the row-column arrangement. In the other eight cases, the computing time limit was reached before the CPLEX solver was able to confirm the optimality of the solution. The computing times for the three simultaneous optimization models are shown in Table 4 as well. The table's final column shows the objective values of the benchmark designs, i.e., the  $W_2$ - and  $W_3$ -optimal row-column arrangements identified by Vo-Thanh et al. (2016).

The results for treatment design 4.1 in the first line of the table correspond to the row-column arrangement in Table 1 and the  $\mathbf{S}_A$  and  $\mathbf{S}_B$  matrices in Table 2. The maximum absolute value of an element of  $\mathbf{S}_A$  and  $\mathbf{S}_B$  for that row-column arrangement is 2, and, in total, there are 24 occurrences of the values +2 or -2, resulting in an objective function value of  $2 \times 10,000 + 24 \times 2 = 20,048$ .

The three simultaneous optimization models result in the best possible objective function value for each two-level treatment design considered. That is not always the case with the sequential approach: for eightand nine-factor treatment designs, and for the designs 7.1, 10.2 and 11.1, the sequential approach results in a larger value for the objective function. For treatment designs 7.1, 10.2 and 11.1, the objective function values obtained from the sequential approach are only 24, 16, and 24 units larger, respectively, than those obtained from the simultaneous approach. This indicates that the maximum absolute elements of the matrices  $\mathbf{S}_A$ and  $\mathbf{S}_B$  are the same for both approaches, and that only the sum of the absolute elements,  $\gamma_{AB}$ , differs to some extent. For treatment designs 8.1, 8.2, 9.1 and 9.2, the objective function value obtained from the sequential approach is about  $20,000 = 2 \times M$  units larger than that obtained from the simultaneous approach. This indicates that the maximum absolute element of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  is different for both approaches: regardless of the exact model used, the simultaneous approach was able to identify row-column arrangements whose  $s_{AB}$  values are two units smaller than those of the row-column arrangements produced by the sequential approach. For all treatment designs with 4–6, 12 and 13 factors, the sequential approach identifies row-column arrangements with the same values for the objective function as the simultaneous approach.

That the sequential approach does not always match the results produced by the simultaneous approach is logical: the sequential approach prioritizes the arrangement of the treatment design in rows, and, therefore, it may result in a poorer subsequent arrangement of the treatment design in columns. It should be pointed out, however, that the sequential approach is faster than the simultaneous approach (regardless of the model used), and that it produces competitive designs, with an  $s_{AB}$  value that is at most two units larger than that produced by the simultaneous approach. When comparing the designs from the sequential approach with the benchmark designs, we can see that their objective function values are alike, so that they possess the same  $s_{AB}$  values for each treatment design considered.

Comparing the row-column arrangements produced by the simultaneous approach and the benchmark row-column arrangements, we can see that their objective function values are identical for seven of the 17 treatment designs studied. For treatment designs 10.1, 10.2, 11.1, 11.2 and 13.2, the benchmark designs have a slightly higher objective function value. This indicates that the maximum absolute elements of the matrices  $\mathbf{S}_A$  and  $\mathbf{S}_B$  are the same for the designs produced by the simultaneous approach and the benchmark designs, and that only the sum of the absolute elements,  $\gamma_{AB}$ , differs to some extent. For treatment designs 7.1, 8.1, 8.2, 9.1 and 9.2, the differences in objective function values are about 20,000, indicating a difference of 2 in the value of  $s_{AB}$  for the row-column arrangements produced by the simultaneous approach and the benchmark row-column arrangements. It is clear, however, that the simultaneous approach leads to rowcolumn arrangements with properties at least similar to those of the benchmark designs produced by the complete enumeration approach of Vo-Thanh et al. (2016).

To support this conclusion, we show the  $A_3^{r,c}$  values of the row-column arrangements produced using our optimization approaches and the benchmark arrangements in Table 4. These values quantify the extent to which two-factor interaction effects are confounded with the rows and the columns. The benchmark designs score best in terms of the  $A_3^{r,c}$  value, because they were selected based on that criterion. For ten of the 17 treatments designs, however, the simultaneous optimization models lead to row-column arrangements with the optimal  $A_3^{r,c}$  value. The sequential approach produces row-column arrangements with an optimal  $A_3^{r,c}$  value for 13 of the 17 treatment designs. For eight-factor and nine-factor treatment designs, there is a substantial difference in  $A_3^{r,c}$  value between the benchmark row-column arrangements and those produced by the simultaneous approach. So, for eight- and nine-factor treatment designs, the difference between the optimization criterion used in the present paper and the criteria used in Vo-Thanh et al. (2016) is largest.

Of the three models we compared for the simultaneous approach, the linearized quadratic model is the fastest for the 17 treatment designs considered. The quadratic programming model is the slowest for each treatment design under consideration. For eight treatment designs, the quadratic programming model was unable to confirm that the solution it returned was indeed optimal. Allowing for more than 10,000 seconds of computing time would remedy this problem. However, the linearized quadratic model did not suffer from this problem and established the optimality of its solutions well within 10,000 seconds. Except for one case, this is also true for the permutation-based model. For this reason, we recommend against using the quadratic model, even though it involves fewer decision variables than the other two simultaneous optimization approaches, fewer constraints than the linearized quadratic programming approach and about as few constraints as the permutation-based approach. The numbers of decision variables and the numbers of constraints in the three simultaneous optimization models are shown in the columns labeled "Var" and "Con" in Table 4.

As a conclusion, our application of the sequential and simultaneous optimization approaches in the 24-run case has shown that the linearized quadratic model is the best simultaneous optimization model, followed by the permutation-based model. The quadratic programming model is not competitive in terms of computing time. Remarkably, in terms of solution quality, the sequential approach performs very well too for the smallest numbers of treatment factors and the largest numbers of treatment factors, even though that approach prioritizes one blocking factor over the other and the comparison we make in this section assumes

Table 4: Results obtained when arranging 24-run two-level treatment designs involving 4–13 factors in four rows and three columns using the sequential approach and the three models for the simultaneous approach. Obj: objective function value; CT: computing time in seconds;  $A_3^{r,c}$ :  $A_3^r + A_3^c$ ; Var: number of decision variables (simultaneous approach only); Con: number of constraints (simultaneous approach only); QM: quadratic model; LQM: linearized quadratic model; PM: permutation-based model.

	, L	Sequentia	l		Simultaneous approach				Donah	morl						
ID		approach	L	Ohi	$\Lambda^{r,c}$		QM			LQM			PM		Denci	шак
	CT	Obj	$A_3^{r,c}$	Obj	A3	CT	Var	Con	CT	Var	Con	CT	Var	Con	Obj	$A_3^{r,c}$
4.1	0.25	20048	0.67	20048	0.67	447.32	253	217	7.27	541	1069	26.55	661	202	20048	0.67
5.1	0.53	40112	1.78	40112	1.78	10000	309	307	28.78	597	1159	69.59	717	293	40112	1.78
6.1	0.58	40168	2.67	40168	2.67	6358.49	379	418	22.06	667	1270	72.88	787	405	40168	2.67
6.2	0.59	40176	2.83	40176	2.83	1896.77	379	418	14.15	667	1270	36.77	787	405	40176	2.83
7.1	1.16	60304	5.44	60280	5.50	4635.45	463	550	11.84	751	1402	66.19	871	538	80280	5.11
7.2	0.44	40248	4.00	40248	4.00	3923.00	463	550	5.51	751	1402	38.47	871	538	40248	4.00
8.1	0.72	80392	7.06	60388	8.61	1939.83	561	703	10.47	849	1555	50.90	969	692	80392	7.06
8.2	0.78	80368	6.61	60412	9.67	10000	561	703	35.62	849	1555	162.49	969	692	80368	6.28
9.1	0.83	80520	10.33	60540	13.00	10000	673	877	15.58	961	1729	98.62	1081	867	80552	10.00
9.2	0.72	80480	9.11	60524	12.11	289.26	673	877	7.54	961	1729	36.47	1081	867	80496	8.78
10.1	0.80	80664	13.00	80664	13.00	10000	799	1072	277.96	1087	1924	346.60	1207	1063	80680	13.00
10.2	1.05	80720	16.00	80704	16.17/16.61	10000	799	1072	18.55	1087	1924	58.30	1207	1063	80744	16.00
11.1	1.22	80896	20.00	80872	20.67	10000	939	1288	20.19	1227	2140	53.15	1347	1280	80912	20.00
11.2	8.56	80992	25.00	80992	25.00	10000	939	1288	1551.49	1227	2140	2046.94	1347	1280	81000	25.00
12.1	11.39	81200	30.00	81200	30.00	10000	1093	1525	2171.92	1381	2377	10000	1501	1518	81200	30.00
13.1	0.77	81040	20.00	81040	20.00	110.78	1261	1783	1.50	1549	2635	8.80	1669	1777	81040	20.00
13.2	0.69	81056	20.00	81056	20.00	30.58	1261	1783	1.00	1549	2635	6.52	1669	1777	81080	20.00

Table 5: Results obtained when arranging 64-run two-level treatment designs of strength 3 involving 6–12 factors in four rows and four columns using the sequential approach, the linearized quadratic model and the permutation-based model. Obj: objective function value; CT: computing time in seconds; LQM: linearized quadratic model; PM: permutation-based model.

	Sequential		Simultaneous approach					
ID	appr	oach	LQM		PM			
	CT	Obj	СТ	Obj	CT	Obj		
6.1	2.29	0	482.14	0	3922.96	0		
7.1	0.73	0	227.34	0	21.40	0		
8.3	0.31	0	3.71	0	4.26	0		
9.1	0.37	0	8.10	0	15.27	0		
9.2	0.36	0	109.04	0	14.98	0		
10.1	0.28	0	3.56	0	14.81	0		
11.1	0.59	0	13.34	0	9.84	0		
11.2	80.25	40064	10.89	0	8.03	0		
11.3	0.20	0	3.40	0	10.23	0		
11.4	107.31	40128	12.46	0	13.04	0		
12.1	40.01	40128	2.57	0	7.89	0		
12.2	94.69	40192	12.95	0	8.44	0		

that the confounding of two-factor interactions with either blocking factor is of equal concern. Due to its speed, we also recommend the sequential approach for small and large numbers of treatment factors.

#### 4.2 64-run designs

In this section, we study the arrangement of 12 strength-3 64-run two-level treatment designs in four rows and four columns, using the sequential approach, the linearized quadratic model and the permutation-based model. We disregarded the quadratic model due to its poor computing times. We focus on scenarios where the optimal value of the objective functions for the sequential approach and the simultaneous approach is known to be zero. So, we study scenarios in which all main effects and all two-factor interaction effects can be made orthogonal to the two blocking factors. The optimal row-column arrangements for these scenarios correspond to the strength-3 64-run mixed-level orthogonal designs of the type  $OA(N, 4^2 \times 2^{q_1}, 3)$  identified by Vo-Thanh et al. (2016). All these row-column arrangements involve up to 12 treatment factors.

For eight of the 12 treatment designs considered, the sequential approach, the linearized quadratic model and the permutation-based model lead to an optimal row-column arrangement, with a zero objective function value. For the four remaining treatment designs, the simultaneous optimization models also produce an optimal row-column arrangement, but the sequential approach does not. So, in these cases, the sequential approach is unable to arrange the rows and the columns orthogonally to all main effects and all two-factor interactions. The treatment designs for which the sequential approach does not result in a zero objective value are all optimally arranged in rows and columns in less than 13 seconds by the linearized quadratic model approach, and in less than 14 seconds by the permutation-based approach. As a result, these particular treatment designs are not very hard to arrange in rows and columns. The exact objective function values and the required computing times for the twelve 64-run treatment designs are shown in Table 5. The permutationbased approach is faster than the linearized quadratic model approach in five of the 12 cases. For treatment design 6.1, however, the permutation-based approach is very slow compared to the linearized quadratic model approach. In terms of the computing time, the linearized quadratic model and the permutation-based model are close competitors for the 64-run designs.

#### 4.3 72-run designs

In this section, we study the arrangement of seven strength-3 72-run two-level treatment designs in three rows and three columns. We again focus on scenarios where the optimal values of the objective functions for the sequential approach and the simultaneous approach are known to be zero. In particular, the optimal

Table 6: Results obtained when arranging 72-run two-level treatment designs of strength 3 involving 6–12 factors in three rows and three columns using the sequential approach, the linearized quadratic model and the permutation-based model. Obj: objective function value; CT: computing time in seconds; LQM: linearized quadratic model; PM: permutation-based model.

	Sequential		Simultaneous approach					
ID	appr	oach	LQM		PM	1		
	CT	Obj	СТ	Obj	CT	Obj		
6.1	1.09	0	60.37	0	1941.45	0		
7.1	88.67	0	10000	40016	10000	40104		
8.1	10000	40048	573.37	0	10000	40328		
9.1	1.11	0	135.88	0	5712.54	0		
10.1	0.92	0	14.62	0	6280.68	0		
11.1	1.09	0	52.57	0	43.41	0		
12.1	0.25	0	104.68	0	48.39	0		

row-column arrangements for these scenarios correspond to the strength-3 72-run mixed-level orthogonal designs of the type  $OA(N, 3^2 \times 2^{q_1}, 3)$  identified by Vo-Thanh et al. (2016). These row-column arrangements involve up to 12 treatment factors.

For five of the seven treatment designs, the sequential approach, the linearized quadratic model approach and the permutation-based model approach succeeded in identifying an optimal row-column arrangement, with an objective function value of zero. In all of these cases, the sequential approach is the fastest. For treatment design 7.1, the simultaneous approach did not produce an optimal row-column arrangement within 10,000 seconds, while the sequential approach was able to find one in about 89 seconds. For treatment design 8.1, the sequential approach and the permutation-based model fail to find an optimal row-column arrangement within 10,000 seconds, while the linearized quadratic model does manage to identify one in less than 600 seconds. In total, the simultaneous approach fails to find two optimal row-column arrangements when the permutation-based model is used. The permutation-based model approach is therefore the poorest of the three approaches investigated in this section in terms of the solution quality for the 72-run designs. For treatment designs 6.1, 9.1, and 10.1, it is also very slow compared to the other two models. Whenever the permutation-based model yields an optimal row-column arrangement fast, the other two methods do not require much computing time either. The exact objective function values and the required computing times for the seven 72-run treatment designs are shown in Table 6.

## 5 Using the sequential approach to create a starting solution for the simultaneous approach

The sequential approach's speed and its excellent performance in terms of the quality of the row-column arrangements suggest that using the row-column arrangement produced by the sequential approach as a starting solution for the simultaneous approach may lead to optimal row-column arrangements in less computing time. This is because many algorithms converge to good solutions more rapidly when provided with a high-quality starting solution. For all the designs problems considered earlier, we explored whether using the output of the sequential approach as input for the simultaneous approach indeed leads to computing time savings. Tables 7, 8 and 9 show our results for the treatment designs involving 24, 64 and 72 runs, respectively. The objective function values for the 24-run arrangements in Table 7 are not shown, because they are the same as those in Table 4.

Comparing the computing times in Table 7, obtained using the output of the sequential approach as input for the simultaneous approach, with those in Table 4 shows that the computing time of the linearized quadratic model goes down for only seven of the 17 treatment designs. For design 12.1, the better starting solution leads to a drop in computing time of more than 12 minutes. In one case, the starting design led to an increase in computing time from about 36 seconds to about 147 seconds. Overall, the changes in computing time for the linearized quadratic model are minor. For the permutation-based approach, the better starting solution led to an improvement in computing time for nine of the 17 treatment designs. The

Table 7: Computing times for the linearized quadratic model (LQM) and the permutation-based model (PM) when arranging 24-run two-level treatment designs involving 4–13 factors in four rows and three columns when using the output of the sequential approach as a starting solution.

ID	LQM	PM
4.1	12.32	65.10
5.1	23.21	99.45
6.1	28.64	60.98
6.2	11.11	24.24
7.1	10.25	61.22
7.2	5.68	23.57
8.1	13.54	40.75
8.2	147.33	174.92
9.1	15.08	123.44
9.2	10.34	32.81
10.1	224.42	281.94
10.2	21.11	96.35
11.1	25.44	84.23
11.2	1448.41	2073.64
12.1	1401.90	4019.71
13.1	2.20	9.59
13.2	1.78	5.40

largest improvement is again obtained for the treatment design with 12 factors. The original simultaneous approach based on the permutation-based model (without high-quality starting solution) was unable to confirm the optimality of the row-column arrangement it found for treatment design 12.1 within 10,000 seconds, while the permutation-based approach involving the better starting solution does establish the optimality of the row-column arrangement in about 4,000 seconds. Comparing the linearized quadratic model with the permutation-based model in Table 7, we can see the former is faster than the latter in all cases.

Comparing the computing times for the 64-run designs in Table 8 with those in Table 5 shows that the computing times for both the linearized quadratic model and the permutation-based model go down for eight of the 12 treatment designs considered. In each of these cases, the sequential approach obtained the optimal solution within a few seconds and the linearized quadratic model and the permutation-based model only need a few fractions of a second extra to confirm the optimality of the solution in terms of the objective function in Equations (18) and (39). For the remaining four treatment designs, the sequential approach fails to find solutions that are optimal in terms of the objective function in Equations (18) and (39) and it uses substantially more computing time. For each of these four cases, the computing times for the linearized quadratic model and the permutation-based model deteriorate much when using the output of the sequential approach as a starting solution. Comparing the linearized quadratic model with the permutationbased model in Table 8, we can see the former is faster than the latter in all but one case (treatment design 12.2).

Comparing the computing times for the 72-run designs in Table 9 with those in Table 6 shows that the computing times of both the linearized quadratic model and the permutation-based model drop substantially (usually by at least one order of magnitude) for six of the seven 72-run treatment designs considered. Again, in each of these cases, the sequential approach obtained the optimal solution within a few seconds, and the linearized quadratic model and the permutation-based model only need a few fractions of a second extra to confirm the optimality in terms of the objective function in Equations (18) and (39). For treatment designs 9.1 and 10.1, the use of the starting solution reduced the computing time for the permutation-based model from about 6,000 seconds to about 1.5 seconds. For treatment design 7.1, the improvement is even more spectacular: the starting solution allows the linearized quadratic model and the permutation-based model to find an optimal row-column arrangement in less than 89 seconds (whereas these models did not allow the simultaneous approach to find the optimal solution within the computing time limit when not using the starting solution provided by the sequential approach). For the remaining treatment design 8.1, the

Table 8: Computing times and objective function values for the linearized quadratic model (LQM) and the permutation-based model (PM) when arranging 64-run two-level strength-3 designs involving 6–12 factors in four rows and four columns when using the output of the sequential approach as a starting solution. CT: computing time in seconds; Obj: objective function value.

	LQI	M	PN	1
ID	CT	Obj	CT	Obj
6.1	2.34	0	2.45	0
7.1	0.78	0	1.08	0
8.3	0.36	0	0.64	0
9.1	0.42	0	0.75	0
9.2	0.41	0	0.78	0
10.1	0.33	0	0.78	0
11.1	0.66	0	1.22	0
11.2	84.57	0	88.90	0
11.3	0.28	0	0.80	0
11.4	109.34	0	112.65	0
12.1	44.80	0	46.85	0
12.2	107.25	0	102.06	0

Table 9: Computing times and objective function values for the linearized quadratic model (LQM) and the permutation-based model (PM) when arranging 72-run two-level strength-3 designs involving 6–12 factors in three rows and three columns when using the output of the sequential approach as a starting solution. CT: computing time in seconds; Obj: objective function value.

obj. objective randtion value.								
ID	LQM	[	PM					
ID	CT	Obj	CT	Obj				
6.1	1.12	0	1.29	0				
7.1	88.70	0	88.92	0				
8.1	13018.32	0	20000	40048				
9.1	1.17	0	1.47	0				
10.1	1.00	0	1.59	0				
11.1	1.17	0	1.90	0				
12.1	0.31	0	1.12	0				

sequential model fails to converge to optimality within 10,000 seconds. Using its output after 10,000 seconds as input for the linearized quadratic model results in an optimal solution after 3,000 more seconds. With the same input, the permutation-based model fails to find an optimal solution within the computing time limit.

## 6 Discussion

In this paper, we proposed several integer programming approaches to arrange a given orthogonal two-level treatment design in rows and columns. This is useful for experiments involving two crossed blocking factors. A major advantage of using integer programming is that, unless the solver is stopped prematurely by the user, it guarantees an optimal solution.

The first approach we presented is a sequential approach, which we originally intended to be used when the confounding of the two-factor interactions with one of the two blocking factors is more of a concern than the confounding with the other blocking factor. However, our computational results show that, for small and for large numbers of treatment factors, the row-column arrangements produced by the sequential approach are also optimal when the confounding of the two-factor interactions with both blocking factors is of equal concern, and that the sequential approach is generally very fast.

The second approach we present is a simultaneous approach which assumes that we are concerned about the confounding of the two-factor interactions with the first blocking factor as much as we are concerned about the confounding of the interactions with the second blocking factor. We describe three variants of the simultaneous approach, one of which, involving a quadratic programming model, requires considerably more computing time than the other two. The latter two variants do not involve quadratic constraints. The so-called linearized quadratic modeling variant is in most cases faster than the permutation-based model. The simultaneous approach outperforms the sequential approach in terms of solution quality for moderate numbers of factors.

We also explored a third kind of approach, in which we use the row-column arrangement produced by the sequential approach as input for the simultaneous approach. This led to very fast computing times for both the linearized quadratic model and the permutation-based model, provided that the sequential approach produces a solution within seconds. Whenever the sequential approach requires 10 seconds or more, using its output as an input for the linearized quadratic model and the permutation-based model slows the solution of these optimization models down. For a few treatment designs, it even causes the solution of the two models to hit the computing time limit.

In the event we are concerned about the confounding of the two-factor interactions with the first blocking factor as much as we are concerned about the confounding of the interactions with the second blocking factor, our recommendation is to use the following procedure for configurations similar to ours:

- 1. Find a row-column arrangement for the treatment design under consideration using the sequential approach.
- 2. If the sequential approach produces an optimal solution within 10 seconds, use that solution as input for the linearized quadratic model.
- 3. If the sequential approach does not finish within 10 seconds, run the linearized quadratic model from scratch, without using a starting solution as input.

We end up recommending the linearized quadratic model rather than the permutation-based model, because the latter occasionally fails to converge to optimality within 10,000 seconds, even though it performs very well in the vast majority of the cases (especially when a high-quality solution from the sequential approach is used as input).

In this article, we applied the various optimization models to sets of 24-run, 64-run and 72-run designs, because high-quality benchmark row-column arrangements exist for these run sizes. We would like to emphasize, however, that both our sequential approach and our simultaneous approach are much more broadly applicable than demonstrated here. More specifically, the high-quality benchmark arrangements were obtained by searching through a complete enumeration of  $OA(N, a \times b \times 2^n, t)$ , while the new approaches presented here work on any single treatment design. This is especially attractive when high-quality treatment designs are readily available, while at the same time a complete enumeration of orthogonal arrays of the type  $OA(N, a \times b \times 2^n, t)$  is infeasible. For example, minimum *G*-aberration designs of strength 3 are known for 32, 40 and 48 runs. However, catalogs of orthogonal arrays of the type  $OA(N, a \times b \times 2^n, 3)$  for these run sizes only cover the following cases: (a)  $OA(32, 4 \times 4 \times 2^n, 3)$  for  $n \le 4$ , (b)  $OA(48, 4 \times 3 \times 2^n, 3)$ for  $n \le 4$  and (c)  $OA(48, 6 \times 4 \times 2^n, 3)$  for  $n \le 2$ . Any strength-2 catalog for  $N \ge 32$  is computationally infeasible. Therefore, in these cases, it is attractive to start with a minimum *G*-aberration design and use our methodology to obtain a good row-column arrangement.

Finally, while we focused on two-level treatment designs here, it is not very difficult to generalize the sequential and simultaneous approaches to deal with multi-level designs and mixed-level designs.

### References

- Cheng, C. S. and Mukerjee, R. (2003). On regular-fractional factorial experiments in row-column designs. Journal of Statistical Planning and Inference, 114:3–20.
- Cheng, S. W. and Wu, C. F. J. (2002). Choice of optimal blocking schemes in two-level and three-level designs. *Technometrics*, 44:269–277.
- Deng, L. Y. and Tang, B. (1999). Generalized resolution and minimum aberration criteria for Plackett-Burman and other nonregular factorial designs. *Statistica Sinica*, 9:1071–1082.
- Gilmour, S. G. and Trinca, L. A. (2003). Row-column response surface designs. Journal of Quality Technology, 35:184–193.
- Goos, P. and Donev, A. N. (2006a). Blocking response surface designs. *Computational Statistics and Data Analysis*, 51:1075–1088.
- Goos, P. and Donev, A. N. (2006b). The D-optimal design of blocked experiments with mixture components. Journal of Quality Technology, 38:319–332.
- Jacroux, M. and SahaRay, R. (1990). On the construction of trend free row-column 2-level factorial experiments. *Metrika*, 37:163–180.
- Jones, B. and Eccleston, J. (1980). Exchange and interchange procedures to search for optimal designs. Journal of the Royal Statistical Society Series B, 42:238–243.
- Sartono, B., Schoen, E. D., and Goos, P. (2015). Blocking orthogonal designs with mixed integer linear programming. *Technometrics*, 57:428–439.
- Schoen, E. D., Eendebak, P. T., and Nguyen, M. V. M. (2010). Complete enumeration of pure-level and mixed-level orthogonal arrays. *Journal of Combinatorial Designs*, 18:123–140.
- Schoen, E. D. and Mee, R. W. (2012). Two-level designs of strength 3 and up to 48 runs. Journal of the Royal Statistical Society Series C, 61:163–174.
- Schoen, E. D., Sartono, B., and Goos, P. (2013). Optimal blocking for general resolution-3 designs. Journal of Quality Technology, 45:166–187.
- Tang, B. and Deng, L. Y. (1999). Minimum G<sub>2</sub>-aberration for nonregular fractional factorial designs. Annals of Statistics, 27:1914–1926.
- Vo-Thanh, N., Schoen, E. D., and Goos, P. (2016). Row-column arrangements of nonregular and regular fractional factorial two-level designs. Working paper, University of Antwerp, Faculty of Applied Economics.
- Williams, E. R. and John, J. A. (1996). Row-column factorial designs for use in agricultural field trials. Journal of the Royal Statistical Society Series C, 45:39–46.
- Wu, C. F. J. and Hamada, M. S. (2009). Experiments: Planning, Analysis, and Parameter Design Optimization. 2nd edition, Wiley, New York, NY, USA.