Construction of Orthogonal Two-level Designs of User-Specified Resolution where \( N \neq 2^k \)

C. T. Liao\(^1\), H. K. Iyer\(^1,2\), and D. F. Vecchia\(^2\)

\(^1\)Department of Statistics, Colorado State University, Fort Collins, CO 80523, USA
\(^2\)National Institute of Standards and Technology, Boulder, CO 80533, USA

Abstract

Fractional 2-level factorial designs are often used in the early stages of an investigation to screen for important factors. Traditionally, \( 2^{n-k} \) fractional factorial designs of resolution III, IV, or V have been used for this purpose. When the investigator is able to specify the set of nonnegligible factorial effects, it is sometimes possible to obtain an orthogonal design with fewer runs than a standard textbook design, by searching within a wider class of designs called parallel-flats designs. The run sizes in this class of designs do not necessarily need to be powers of 2. We discuss an algorithm for constructing orthogonal parallel-flats designs to meet user specifications. Several examples illustrate the use of the algorithm.

Key Words: Parallel flats designs; Single-flat designs; Fractional factorial designs; Screening experiments; Linear graphs.

1 Introduction

During the initial stages of a product or a process design, design engineers typically consider several factors which may have an influence on a performance measure (or measures) of interest. Some of the factors may affect the response to a greater extent than others. In trying to understand the relative importance of each factor, it is often desirable to run one or more screening experiments to identify the more influential factors. Once the influential factors have been identified, and confirmed, one can plan more elaborate experiments to characterize the relationship between the response variable(s) and the experimental factors by means of suitable mathematical functions.

Factorial designs where each factor has 2 or 3 levels are most suited for factor screening. The total number of treatment combinations in a complete factorial experiment is kept relatively low by such designs and, when the number of factors is not too large, a complete factorial design may offer an economically feasible option. When the number of factors is large, cost considerations may lead the experimenter to use fractional factorial designs.
Before selecting a fractional factorial design for a particular application, it is good practice to partition the full set of factorial effects into the following three disjoint sets.

1. Primary effects: factorial effects for which the experimenter wishes to obtain (unbiased) estimates. Denote this set of effects by \( G_1 \).

2. Secondary effects: Possibly nonnegligible factorial effects for which the experimenter is not interested in obtaining unbiased estimates at this stage of experimentation; such effects must be included in the model. Denote this set of effects by \( G_2 \).

3. Negligible effects: The remaining factorial effects thought by the experimenter to be negligible. Denote this set of effects by \( G_3 \).

This partitioning of factorial effects has been suggested by Franklin and Bailey (1977). It is very easy to accommodate blocking requirements as long as the number of blocks is a power of 2. If we want an orthogonal design for \( n \) factors in \( 2^r \) blocks then we introduce \( r \) 2-level pseudo factors \( F_{n+1}, \ldots, F_{n+r} \) and include the main effects and interactions of all orders among pseudo-factors in the set \( G_2 \).

Typically, the set of primary effects would consist of all main effects and some low order interactions, whereas high order interactions would form the set of negligible effects. Any remaining effects would be secondary effects. It is then of interest to find designs using which one can obtain unbiased estimates of all effects in \( G_1 \) based on a factorial linear model in which the effects in \( G_3 \) are assumed to be zero. Any such design will be termed a design of resolution \( (G_1,G_2) \).

We define \( FFD(2^n, N, G_1, G_2) \) to be the set of all \( N \)-run designs involving \( n \) 2-level factors and having resolution \( (G_1,G_2) \). The full \( 2^n \) factorial design belongs to the set \( FFD(2^n, 2^n, G_1, G_2) \) for any allowable choice of \( G_1 \) and \( G_2 \). If \( G_1 \) consists of the general mean \( \mu \) and all main effects, and \( G_2 \) is the empty set, then \( FFD(2^n, N, G_1, G_2) \) is in fact the set of all \( N \) run resolution III designs for \( n \) 2-level factors. If \( G_1 \) is as above but \( G_2 \) consists of all two-factor interactions, then \( FFD(2^n, N, G_1, G_2) \) is the set of all \( N \) run resolution IV designs for \( n \) 2-level factors. Similar comments apply to designs of resolution V, VI, VII, etc.

In this paper we present an algorithm for constructing orthogonal 2-level fractional factorial designs of user-specified resolution \( (G_1,G_2) \) with relatively small numbers of runs. The method also allows for blocking, but the number of blocks must be a power of 2. Designs produced by the algorithm need not have the number of runs equal a power of 2. We give examples where the algorithm produces orthogonal designs having a smaller number of runs than standard textbook designs.
2 Preliminaries

We first describe the notation used in this paper. We use $F_1, \ldots, F_n$ to denote the $n$ 2-level factors. Also we use $-1$ to denote the low level and $+1$ to denote the high level of each factor. As is common practice, $F_1, \ldots, F_n$ will also represent the main effects. The expression $F_1^{e_1}F_2^{e_2}\ldots F_n^{e_n}$ will represent a general factorial effect with $e_i$'s being 0 or 1. If $e_i$ is 1 then $F_i$ appears in the factorial effect, otherwise it does not. The vector $e = (e_1, \ldots, e_n)$ is called the defining vector for the general factorial effect $F_1^{e_1}F_2^{e_2}\ldots F_n^{e_n}$. The defining vector $e = (0, 0, \ldots, 0)$ is for the grand mean $\mu$. A treatment combination or run will be represented by an $n$-tuple whose entries are $-1$ or $+1$ depending on whether the corresponding factor occurs at the low level or at the high level, respectively.

We now review definitions of fractional factorial designs and orthogonal designs.

Fractional Factorial Designs:

When the number of factors $n$ is large, it is uneconomical to conduct an experiment which uses all possible treatment combinations. Traditionally, a fractional factorial design with $2^{n-k}$ runs is used for a suitably chosen $k$. Classical fractional factorial designs are typically described using appropriate defining relations. For instance, consider the following design for 7 factors in 8 runs.

<table>
<thead>
<tr>
<th>run</th>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$</th>
<th>$F_4$</th>
<th>$F_5$</th>
<th>$F_6$</th>
<th>$F_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
</tr>
<tr>
<td>5</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>6</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>7</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>8</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

This design is defined using the relations $F_1F_3F_6 = +I$, $F_2F_4F_7 = -I$, $F_3F_5F_7 = -I$, and $F_4F_5F_6F_7 = +I$.

In theoretical investigations, it is useful to represent designs and defining relations using 0 in place of $+1$ and 1 in place of $-1$. This allows for arithmetical operations modulo 2 and
the use of the Galois Field GF(2). With this choice of notation the design above becomes

<table>
<thead>
<tr>
<th>run</th>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$</th>
<th>$F_4$</th>
<th>$F_5$</th>
<th>$F_6$</th>
<th>$F_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

and the defining relations can be expressed in a matrix form

$$At = c$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}, \quad t = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_7 \end{bmatrix}, \quad c = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

(2.1)

and $(t_1, t_2, \cdots, t_7)$ is a treatment vector in the design. Using geometric terminology, such designs have been called single-flat designs. $A$ is called the alias matrix and $c$ is called the coset indicator vector.

A $2^{n-k}$ fractional factorial forms a $1/2^k$ replicate of the full $2^n$ factorial design. Some authors have considered the use of $j/2^k$ replicates (see for instance, Kempthorne (1975), page 420). Using geometric terminology, such designs have been called parallel-flats designs. They are simply the union of several fractional factorial designs which use the same defining relations except for the choice of $\pm I$. The following example illustrates such a design.

**Example 1:** The union of the treatment combinations in the following three fractional factorial designs forms a parallel-flats design consisting of 3 flats.

**Design 1:** $F_1 F_2 F_3 = +I$, $F_2 F_3 F_7 = -I$, $F_3 F_6 F_7 = -I$, and $F_1 F_5 F_6 F_7 = +I$.

**Design 2:** $F_1 F_5 F_6 = +I$, $F_2 F_5 F_7 = -I$, $F_3 F_6 F_7 = -I$, and $F_4 F_5 F_6 F_7 = -I$.

**Design 3:** $F_1 F_5 F_6 = -I$, $F_2 F_5 F_7 = +I$, $F_3 F_6 F_7 = -I$, and $F_4 F_5 F_6 F_7 = -I$.

This is in fact a $3/16$ replicate of a $2^7$ factorial design. Using 0's and 1's (mod 2) rather than $\pm 1$, the treatment combinations for the design $i$, denoted by $D_i$, satisfy the equation

$$At = c_i$$

where $A$ is as in (2.1). $c'_1 = [0 \ 1 \ 1 \ 0]$, $c'_2 = [0 \ 1 \ 1 \ 1]$, and $c'_3 = [1 \ 0 \ 1 \ 1]$. Note that $A$ is the same for each component design. The parallel-flats design consists of the following
treatment combinations.

<table>
<thead>
<tr>
<th>run</th>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$</th>
<th>$F_4$</th>
<th>$F_5$</th>
<th>$F_6$</th>
<th>$F_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>17</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

The first 8 runs belong to Design 1, the second 8 runs to Design 2, and the final 8 runs to Design 3, each of which is a $2^{7-4}$ design. The matrix $A$ is called the alias matrix as before and the matrix $C = [c_1:c_2:c_3]$ is called the coset indicator matrix.

It may be of interest to note that many of the published orthogonal main effect plans with run sizes not equal to a power of 2 can be shown to be parallel-flats designs. This is what motivated us to search in this wider class for designs with user-specified resolution.

**Orthogonal Designs:**

Let $\beta$ denote the vector of factorial effects that are not assumed to be zero. The elements of $\beta$ belong to either $G_1$ or $G_2$. The corresponding linear model for the observations from an experiment using a design $D$ may be written in the form

$$Y = X\beta + \epsilon \quad (2.2)$$

where the matrix $X$ depends on the design $D$ and is often called the design matrix. The
vector

\[ \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} \]

consists of all factorial effects in \( G_1 \cup G_2 \), with \( \beta_1 \) consisting of all factorial effects in \( G_1 \) and \( \beta_2 \) consisting of all factorial effects in \( G_2 \). The vector \( \epsilon \) consists of noise random variables which are assumed to have zero means and are pairwise uncorrelated with a common variance \( \sigma^2 \) (usually unknown). Let the matrix \( X \) be written as \([X_1 : X_2]\) with \( X_1 \) corresponding to \( \beta_1 \) and \( X_2 \) corresponding to \( \beta_2 \). The design \( D \) is said to be orthogonal for \( \beta_1 \) if the matrix \( X_1'X_1 \) is a diagonal matrix and the matrix \( X_1'X_2 \) is zero.

Orthogonal designs possess many important properties. It is well known that if \( D \) is an orthogonal design for \( \beta_1 \) with \( N \) runs, it is optimal (A-optimal, D-optimal, E-optimal; see Kiefer (1959)) for estimating \( \beta_1 \) in the class \( FFD(2^n, N, G_1, G_2) \). Orthogonal designs also lead to simple interpretation of results. For these reasons orthogonal designs are very popular. However they may not exist for every value of \( N \). The subject of existence and construction of orthogonal designs has been studied by many authors during this century. Next, we discuss some historical background concerning orthogonal designs.

3 Historical Background

C.R. Rao (1950) derived conditions on the alias matrix \( A \) for a single-flat design determined by \( A \) and a coset indicator vector \( c \) to be an orthogonal design of resolution \( d \). His results are applicable to \( s^n \) factorial experiments where \( s \) is any prime number or a power of a prime number.

The class of parallel-flats designs was introduced by Connor and Young (1961). Srivastava and Chopra (1973) stated the necessary and sufficient conditions for a parallel-flats design to be an orthogonal design of resolution \( d \). A proof of their results can be found in Srivastava and Throop (1990). A general theory of parallel-flats designs was presented in Srivastava, Anderson, and Mardekan (1984), and was later extended in Srivastava (1987). Based on these results, Li (1991) obtained necessary and sufficient conditions for a parallel-flats design to be an orthogonal design for estimating a specified set of factorial effects. Li's result was developed for the case of \( s^n \) factorial experiments with \( s \) a prime number. If applied directly, her results pertain to the case of designs of resolution \((G_1, G_2)\) with \( G_2 \) being the empty set. However, the method of her proof can be easily applied to extend the result to nonempty sets \( G_2 \). Li (1991) also gave several orthogonal designs for \( 2^n \) experiments as an application of her theorem. Some of the designs she provided have run sizes that are not powers of 2.

The literature on general methods for constructing orthogonal 2-level factorial designs for estimating a user-specified set of factorial effects is not as rich. Greenfield (1976) seems to have been the first one to address this problem from an algorithmic point of view. His algorithm was later generalized by Franklin and Bailey (1977), who showed how to incorporate blocking requirements. In essence, their algorithm is a systematic, exhaustive search
for all $2^{n-k}$ designs, for a given $k$, that are of user-specified resolution ($G_1, G_2$). If their
algorithm fails to find a solution then it follows that a design with the specified requirements
does not exist. Franklin (1985) extended the algorithm to include $s^k$ factorials, $s$ being a
prime number, and allowed for blocking. The FACTEX procedure of SAS/QC® software
implements the Franklin-Bailey algorithm as does the RS/Discover® software package. The
main drawback of the Franklin-Bailey (1977) and Franklin (1985) algorithms seems to be
that they quickly become computationally infeasible as the number of factors grows. Of
course, the computational limits of their algorithm will depend on the type of hardware
used. No detailed information is available regarding the current limitations of the various
algorithms.

A different approach to finding orthogonal designs for estimating a specified set of factorial
effects has been adopted by other authors. According to their approach, one first produces
a table of orthogonal designs for various numbers of factors and run sizes. Then the alias
structure of each tabulated design is identified. The final step consists of comparing the
user-specified set of factorial effects with the tabulated collection of designs until a match
is found; that is, by suitably naming the factors in a tabulated design, all effects of interest
are discovered to be estimable, assuming higher order factorial effects to be zero. Some of
the earliest published works using this approach are by Taguchi (1959, 1960). Other papers
which follow a similar approach include Li, Washio, Iida, and Tanimoto (1990), Kacker and
Tsui (1990), and Wu and Chen (1992). Among these, the work of Wu and Chen seems to be
the most general. They discuss a graph-aided method and suggest an approach for computer
implementation of their method. In principle, their method can be used to obtain designs of
resolution ($G_1, G_2$) as long as neither $G_1$ nor $G_2$ contains any interaction involving 3 or more
factors. Again, no detailed information is available regarding the computational limits of
their procedure, but it appears that their method is presently not feasible when the number
of factors is large; Wu and Chen (1992) do not discuss computational feasibility.

Each of the computational methods discussed so far will only yield designs with run sizes
that are powers of 2, since all are based on single-flat designs. In this paper, we propose
an algorithm which searches for an orthogonal design of resolution ($G_1, G_2$) in the class of
parallel-flats designs. Consequently, our algorithm is capable of finding suitable designs with
run sizes not necessarily powers of 2. This means that in some instances we can find a
smaller design for estimating a given set of factorial effects than can be found by the other
algorithms. Our algorithm is capable of incorporating blocking requirements. Even though
the number of blocks must be a power of 2, the sizes of the blocks need not be powers
of 2. Unlike the graph-aided algorithm of Wu and Chen, our algorithm does not impose
restrictions on the types of factorial effects that can be in $G_1$ or $G_2$. Although we cannot
guarantee that the algorithm will find the minimum run design for each given problem, in
nearly all of the tests conducted so far it has produced an orthogonal design with run size
equal to or smaller than various published designs for estimating the same set of factorial
effects.

In the next section we state the algorithm and give an example illustrating how it actually
works. In section 5 we illustrate the capabilities of the procedure by providing several designs
obtained from a computer program based on the algorithm.
Before describing the algorithm, we summarize the definitions and notation used in this paper in the following table.

<table>
<thead>
<tr>
<th>$G_1$</th>
<th>Set of primary factorial effects which we wish to estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_2$</td>
<td>Set of factorial effects that are not of direct interest but cannot be assumed equal to zero.</td>
</tr>
<tr>
<td>$G_3$</td>
<td>Set of factorial effects that are assumed equal to zero.</td>
</tr>
<tr>
<td>A design of resolution $(G_1, G_2)$</td>
<td>Any design using which we can obtain unbiased estimates of all effects in $G_1$ under a linear model in which all factorial effects not contained in $G_1 \cup G_2$ are assumed to be zero.</td>
</tr>
<tr>
<td>$FFD(2^n, N, G_1, G_2)$</td>
<td>The set of all fractional factorial designs for $n$ 2-level factors using $N$ runs, having resolution $(G_1, G_2)$.</td>
</tr>
<tr>
<td>$e_1 \oplus e_2$</td>
<td>Componentwise addition of vectors modulo 2.</td>
</tr>
</tbody>
</table>

### 4 An Algorithm

The algorithm we propose is based on an expression for the general element of the information matrix $X'X$ of an arbitrary parallel-plats design, where $X$ is the design matrix in the linear model $Y = X\beta + \epsilon$. We state the result as a formal proposition. For a proof, refer to Liao(1994).

**Proposition 4.1.**

Let $D$ be the parallel-plats design determined by the alias matrix $A$, which is $k \times n$ of rank $k$, and the coset indicator matrix $C$ ($k \times f$). Let $B$ be a $n \times (n-k)$ matrix of rank $(n-k)$ such that $AB = 0$ and $Z$ be a $n \times f$ matrix such that $AZ = C$. Suppose the linear model in (2.1) holds. The element of the information matrix $M = X'X$ corresponding to the row indexed by the factorial effect whose defining vector is $e_1 = (e_{11}, \ldots, e_{1n})$ and the column indexed by the factorial effect whose defining vector is $e_2 = (e_{21}, \ldots, e_{2n})$ is $m(e_1, e_2)$ given by

$$m(e_1, e_2) = [\sum_{i=1}^{f}(-1)^{\langle e_1 \oplus e_2 \rangle Z_i}] [\sum_{v}(-1)^{\langle e_1 \oplus e_2 \rangle Bv}]$$  \hspace{1cm} (4.1)

where $v$ is an element of $GF_2^{n-k}$ ($GF_2^n$ is the set of all binary vectors of length $r$). The exponents of $-1$ in both factors of (4.1) are computed using arithmetic modulo 2 but the sums of the powers of $-1$ are not performed modulo 2.

The expression in (4.1) implies the following. Suppose $e_1$ and $e_2$ are defining vectors for two distinct factorial effects in $G_1$, or one in $G_1$ and one in $G_2$. For orthogonality, we want $m(e_1, e_2) = 0$. This requires that one of the two terms on the right hand side of (4.1) must be zero. The term $[\sum_{v}(-1)^{\langle e_1 \oplus e_2 \rangle Bv}]$ is zero precisely when the factorial effects
corresponding to \( e_1 \) and \( e_2 \) are not aliased with each other under the alias matrix \( A \), i.e., the vector \( e = e_1 \oplus e_2 \) is not in the row space of \( A \). If, however, \( e_1 \) and \( e_2 \) are aliased under \( A \), the coset indicator matrix \( C \) must be chosen such that the term \( \sum_{i=1}^{f} (-1)^{(e_i \oplus e_2)z_i} \) is zero, i.e., \( z_1, z_2, \ldots, z_f \) are chosen such that the vector \( ez = (ez_1, ez_2, \ldots, ez_f) \) contains the same number of 0's and 1's. We illustrate with an example.

**Example 2:** In a \( 2^6 \) experiment, suppose \( G_1 = \{ \mu, F_1, F_2, F_3, F_4, F_5, F_6, F_1F_2, F_1F_3, F_4F_5 \} \), \( G_2 = \phi \) (empty set). Consider a \( 2^{6-4} \) fractional factorial design defined by

\[
F_2 = I, \quad F_3 = I, \quad F_1F_4 = I, \quad F_5 = I
\]

Here,

\[
A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
1 & 1 \\
0 & 0 \\
0 & 0 \\
1 & 1 \\
0 & 0 \\
0 & 1
\end{bmatrix}
\]

It is easily verified that \( \sum (-1)^{(e_i \oplus e_2)Bv} \neq 0 \) for the following pairs of effects in \( G_1 \):

\[
(F_2, \mu), (F_3, \mu), (F_5, \mu), (F_1, F_4), (F_1, F_1F_2), (F_1, F_1F_3),
\]

\[
(F_2, F_3), (F_2, F_5), (F_3, F_5), (F_4, F_1F_2), (F_4, F_1F_3), (F_1F_2, F_1F_3)
\]  \( (4.2) \)

and is 0 for all other pairs. If the vectors \( z_1, z_2, z_3 \), and \( z_4 \) are chosen as the columns of

\[
Z = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} = [z_1, z_2, z_3, z_4]
\]

(so \( c_1 = (0, 0, 0, 0)' \), \( c_2 = (1, 0, 1, 1)' \), \( c_3 = (0, 1, 1, 1)' \), and \( c_4 = (1, 1, 0, 0)' \)), then the first term in \( (4.1) \) is zero for the pairs in \( (4.2) \).

This example suggests the following algorithm.

1. Select a suitable alias matrix \( A \); that is, define a \( 2^{n-k} \) fractional factorial design.

2. Identify pairs of effects for which the second term in \( (4.1) \) is nonzero. Choose \( z_1, z_2, \ldots, z_f \) such that the first term in \( (4.1) \) is zero for these pairs.

The resulting design using \( A \) and \( C \) (recall \( C = AZ \)) is an orthogonal design, as desired.

An alternate, or dual, approach would be to first choose \( z_1, z_2, \ldots, z_f \) in a judicious manner and identify pairs of effects for which the first term in \( (4.1) \) is nonzero. Then find a
$2^{n-k}$ fractional factorial design such that these pairs of effects are unaliased, so the second term in (4.1) is zero.

The dual approach seems to offer certain advantages over the first approach. Often vectors $z_1, z_2, \ldots, z_f$ can be chosen such that there are only a small number of effects pairs $(e_1, e_2)$ for which $m(e_1, e_2)$ is not zero. We can now use the well-known Franklin-Bailey algorithm on a smaller problem and obtain an alias matrix $A$ for which the second term in (4.1) is zero for these pairs. We illustrate the dual approach for the preceding example.

**Example 2 (continued):** We choose

$$Z = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} = [z_1, z_2, z_3, z_4]$$

The pairs of effects in $G_1$ for which the first term in (4.1) is nonzero are

$$(F_1, F_6), (F_3, F_4), (F_2, F_5), (F_1 F_2, F_3), (F_1 F_2, F_4), (F_1 F_3, F_2), (F_1 F_3, F_5), (F_2, F_4 F_5), (F_5, F_4 F_5), (F_1 F_3, F_4 F_5)$$

We seek a $2^{6-4}$ design (if one exists) in which the above pairs of effects are not confounded. Such a design may be found using the Franklin-Bailey algorithm. One particular solution has an alias matrix given by

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

corresponding to the defining relations $I = F_2 F_3 = F_1 F_4 = F_5 = F_6$.

Finding an orthogonal design of resolution $(G_1, G_2)$ with the fewest number of runs generally requires an exhaustive search approach. This is feasible in "small" problems, but quickly becomes infeasible as the number of factors grows. Our focus is on finding "small" designs satisfying the requirements, though not necessarily having minimum number of runs. In the next subsection we explain our algorithm and give some examples.

### 4.1 Statement of the Algorithm

Specification of $G_1$ and $G_2$ is equivalent to specification of factorial effects that should not be aliased with $\mu$. Franklin and Bailey (1977) refer to these as ineligible effects. The set of such effects consists of every nontrivial effect (i.e., any effect other than $\mu$) in $G_1$ along with effects that are products (generalized interactions) of pairs of distinct effects, one from $G_1$
and the other from either \( G_1 \) or \( G_2 \). The set of defining vectors corresponding to this set of ineligible factorial effects is denoted by \( \mathbf{R} \). For instance, if \( G_1 = \{ \mu, F_1, F_2, F_3, F_4 \} \) and \( G_2 = \{ F_1 F_2, F_1 F_3 F_4 \} \), then the set \( \mathbf{R} \) consists of defining vectors corresponding to the effects \( F_1, F_2, F_3, F_4, F_1 F_2, F_1 F_3, F_1 F_4, F_2 F_3, F_2 F_4, F_3 F_4, F_1 F_2 F_3 F_4, F_1 F_2 F_3, F_1 F_2 F_4 \). Thus

\[
\mathbf{R} = \{(1,0,0,0),(0,1,0,0),(0,0,1,0),(1,0,0,1),(1,1,0,0),(0,0,0,1),
(0,1,1,0),(0,1,0,1),(0,0,1,1),(1,0,1,1),(1,1,1,0),(1,1,0,1)\}
\]

However, if \( G_1 = \{ F_1, F_2, F_3, F_4 \} \) and \( G_2 = \{ \mu, F_1 F_2, F_1 F_3 F_4 \} \), then the set \( \mathbf{R} \) consists of defining vectors corresponding to the effects \( F_1, F_2, F_3, F_4, F_1 F_2, F_1 F_3, F_1 F_4, F_2 F_3, F_2 F_4, F_3 F_4, F_1 F_2 F_3 F_4, F_1 F_2 F_3, F_1 F_2 F_4 \). Thus

\[
\mathbf{R} = \{(1,0,0,0),(0,1,0,0),(0,0,1,0),(0,0,0,1),(1,1,0,0),(1,0,1,0),
(1,0,0,1),(0,1,1,0),(0,1,0,1),(0,0,1,1),(1,1,1,0),(1,1,0,1)\}
\]

It is convenient to state the implication of equation (4.1) in terms of the set \( \mathbf{R} \). We get the following proposition.

**Proposition 4.2:**

Let \( D, A, C, B, \) and \( Z \) be as in Proposition 4.1. Let \( G_1 \) and \( G_2 \) be the sets of primary effects and secondary effects, respectively, and let \( \mathbf{R} \) be the set of effects that should be unaliased with \( \mu \). The design \( D \) is an orthogonal design of resolution \( (G_1,G_2) \) if and only if, for each vector \( e \) in \( \mathbf{R} \), either

\[
\sum_{i=1}^{f} (-1)^{e} z_i = 0 \tag{4.3}
\]

or

\[
\sum_{v} (-1)^{e} B v = 0 \tag{4.4}
\]

where \( v \) is an element of \( GF_2^{m-k} \).

The main idea behind our algorithm is based on the observation that in many practical problems it is possible to choose a \( n \times f \) matrix \( Z \) with columns \( z_1, z_2, \ldots, z_f \) such that the term in (4.3) is zero for most of the effects in \( \mathbf{R} \) excepting those in a subset \( \mathbf{R}_0 \) of \( \mathbf{R} \). We can then use the Franklin and Bailey (1977) algorithm, with the reduced set \( \mathbf{R}_0 \) of ineligible effects, to find an alias matrix \( A \) (equivalently, \( B \)) using which the term in (4.4) is zero for every effect in \( \mathbf{R}_0 \). Thus \( A \) and \( C \) (i.e., \( AZ \)) together define a parallel-flats design that is orthogonal of resolution \( (G_1,G_2) \).

This procedure will always yield an orthogonal parallel-flats design regardless of how \( Z \) is selected. However a judicious choice of \( Z \) often will produce a design with fewer runs than what an arbitrary choice of \( Z \) might. Since most screening experiments primarily are concerned with main effects and perhaps a few higher order interactions, it would be logical to select a \( Z \) which makes use of this fact. Our experience shows that \( Z \) often can be taken to
be a matrix having rows selected from a Hadamard matrix of suitable size. (For information about Hadamard matrices see Hedayat and Wallis (1978)). Such a choice automatically forces the term in (4.3) to be zero for most of the effects in $R$. Consequently, the subset $R_0$ is much smaller than the set $R$ and the exhaustive Franklin-Bailey algorithm can be used to find a suitable $A$ matrix. We now give the details of our algorithm based on the principles above.

Suppose we are attempting to obtain a design with $N = f \times 2^{n-k}$ runs which is orthogonal of resolution $(G_1, G_2)$. We assume that $f$ is a multiple of 4 so that $f \times f$ Hadamard matrices (equivalently $f$-run Plackett Burman designs) may be used in our construction. (This restriction is not necessary if one wants to use other approaches for choosing $Z$). We use 0's in place of +1's and 1's in place of -1's when writing Hadamard matrices. The algorithm we propose consists of the following steps.

Step (1): Select a $n \times f$ matrix $Z$ using which the expression in (4.3) is zero for a large number of effects in $R$. Calculate the subset $R_0$ of $R$ consisting of effects for which (4.3) is nonzero.

Comments: Our implementation of the algorithm uses the following heuristic approach for selecting $Z$. The rows of $Z$ are chosen from a Hadamard matrix of size $f \times f$. Not all rows of the Hadamard matrix need appear in $Z$ and some rows of the Hadamard matrix may appear more than once. The selection of rows from the Hadamard matrix is essentially based on the objective that we would like the set $R_0$ to be "small". Recall that we will always get an orthogonal design of the type desired regardless of how $Z$ is chosen, but the choice of $Z$ can affect the size of the final design. The details of our heuristic approach are as follows.

Heuristic choice of $Z$:

(a) First determine the total number of times each factor appears as a component of an effect in $R$, and order the factors from the largest to the smallest frequency of occurrence.

(b) The first $i$ factors (we take $i$ to be 0, 1, 2, 3 or 4) are then assigned the first row $h_1$ (of all zeros) from $H$ (a Hadamard matrix). In many problems, assigning factors to the row of all zeros will substantially reduce the set of ineligible effects $R_0$. This is particularly so when the number of interactive factors is small.

(c) At this point several options are available for assigning rows of $H$ to the remaining factors. In principle, one could find an assignment that would make the set $R_0$ as small as possible, but this would involve substantial computation. Another option would be to randomly assign rows of $H$ to the remaining factors. In this case, several random assignments may be tried. Rather than use either of these assignment methods, we have chosen to use two systematic strategies of assignment which seem to work satisfactorily. These two methods are as follows.

The first strategy is to partition the remaining factors into groups of almost equal size. This leads to at most $f - 1$ groups of factors with each group containing $\lfloor \frac{f-1}{4} \rfloor$ or $\lceil \frac{f+1}{4} \rceil + 1$
factors (here, $\lfloor x \rfloor$ stands for the greatest integer not greater than $x$). Each group of factors is assigned a distinct row of $H$. We call this strategy "group" assignment.

The second strategy that often is successful is as follows. Remaining factors (the ones not already assigned to $h_1$) are assigned in order of decreasing frequency to distinct nonzero rows of $H$, say $h_2, \ldots, h_f$, until all $f - 1$ rows are used. At this point, we cycle back to $h_2$ and continue the assignment, in order, to other unassigned factors. We call this strategy "cyclic" assignment.

Although these two strategies are not directly derived from the objective of reducing the size of the set $R_0$, a computer program using these two strategies has been reasonably successful in finding small, parallel-flats orthogonal designs in many situations.

**Step (2):** Use the exhaustive Franklin-Bailey algorithm to find a matrix $A$ of size $k \times n$ of rank $k$ (if one exists), using $R_0$ as the set of ineligible effects. If successful, the expression in (4.4) is guaranteed to be zero for every effect in $R_0$.

**Step (3):** Calculate $C = AZ$. Construct a parallel-flats design using $A$ and $C$. The result is an orthogonal design of user-specified resolution $(G_1, G_2)$.

**Step (4):** If in step (2), the Franklin-Bailey algorithm fails to find a design of the required type, one may go back to step (1) and choose a different $Z$ and repeat the process. The process may be repeated a convenient number of times until the algorithm either produces a suitable design or fails to find such a design. In our implementation we try different values for $i$ in step (1), specifically $i = 0, 1, 2, 3,$ or $4$, and also try both the "group" strategy and the "cyclic" strategy. Thus, we try 10 different $Z$ matrices. If a design has not been found after 10 attempts, we begin a new search for a design with a larger number of runs. Eventually the algorithm will lead to an orthogonal design of resolution $(G_1, G_2)$.

We give two detailed examples to illustrate how the algorithm works. The examples also illustrate the two strategies for selecting the rows of the $Z$ matrix from rows of a chosen Hadamard matrix.

**Example 3:** Consider a $2^{10}$ factorial experiment in which we need to estimate all 10 main effects. Suppose $F_7 F_8, F_7 F_{10}, F_8 F_{10}$ may be nonnegligible (factors $F_7, F_8,$ and $F_{10}$ may interact among themselves), but we are not interested in estimating the associated effects at this point. The remaining interactions are assumed to be zero.

Suppose we wish to find a design with 16 runs in 4 blocks to meet the above requirements. We introduce two pseudo-factors, say $F_{11}$ and $F_{12}$, and put $F_{11}, F_{12},$ and $F_{11} F_{12}$ into $G_2$. We have

$$G_1 = \{\mu, F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9, F_{10}\}$$

and

$$G_2 = \{F_7 F_8, F_7 F_{10}, F_8 F_{10}, F_{11}, F_{12}, F_{11} F_{12}\}.$$
Step (1): In this case, it is easy to check that the set \( R \) consists of 110 elements, but we do not list them all here. We start with the \( 4 \times 4 \) Hadamard matrix below (note that +1's and -1's have been replaced by 0's and 1's, respectively).

\[
H = \begin{bmatrix}
  h_1 \\
  h_2 \\
  h_3 \\
  h_4 \\
\end{bmatrix} = \begin{bmatrix}
  0 & 0 & 0 & 0 \\
  0 & 1 & 0 & 1 \\
  0 & 0 & 1 & 1 \\
  0 & 1 & 1 & 0 \\
\end{bmatrix}
\]

The frequency of occurrence in \( R \) and the assignment for each factor obtained from the "group" strategy are as follows.

<table>
<thead>
<tr>
<th>Factor</th>
<th>( F_7 )</th>
<th>( F_8 )</th>
<th>( F_{10} )</th>
<th>( F_{11} )</th>
<th>( F_{12} )</th>
<th>( F_1 )</th>
<th>( F_2 )</th>
<th>( F_3 )</th>
<th>( F_4 )</th>
<th>( F_5 )</th>
<th>( F_6 )</th>
<th>( F_9 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>26</td>
<td>26</td>
<td>26</td>
<td>20</td>
<td>20</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>when ( i = 0 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_4 )</td>
<td>( h_4 )</td>
</tr>
<tr>
<td>when ( i = 1 )</td>
<td>( h_1 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_4 )</td>
<td>( h_4 )</td>
</tr>
<tr>
<td>when ( i = 2 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_4 )</td>
<td>( h_4 )</td>
</tr>
<tr>
<td>when ( i = 3 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_4 )</td>
<td>( h_4 )</td>
</tr>
<tr>
<td>when ( i = 4 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_1 )</td>
<td>( h_2 )</td>
<td>( h_2 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_3 )</td>
<td>( h_4 )</td>
<td>( h_4 )</td>
</tr>
</tbody>
</table>

For this example, the assignment corresponding to \( i = 2 \) in the table leads to the desired design. The corresponding \( Z \) matrix is given below.

\[
Z = \begin{bmatrix}
  0 & 0 & 1 & 1 \\
  0 & 0 & 1 & 1 \\
  0 & 0 & 1 & 1 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 0 \\
  0 & 0 & 0 & 0 \\
  0 & 0 & 0 & 0 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 0 & 1 \\
  0 & 1 & 0 & 1 \\
  0 & 1 & 0 & 1 \\
\end{bmatrix}
\]

The set \( R_0 \) corresponding to this choice of \( Z \) consists of the defining vectors corresponding the effects

\[
F_7, F_8, F_1 F_2, F_1 F_3, F_2 F_3, F_4 F_5, F_4 F_6, F_4 F_9, F_5 F_6, F_5 F_9, F_6 F_9,
F_7 F_8, F_{10} F_{11}, F_{10} F_{12}, F_{11} F_{12}, F_7 F_{11} F_{12}, F_8 F_{11} F_{12}.
\]

Step (2): Now we apply the Franklin-Bailey algorithm, using the ineligible effects in \( R_0 \), to find a 4-run fractional factorial design. One solution is given by the defining relations

\[
I = F_3 = F_1 F_4 = F_1 F_5 = F_2 F_6 = F_1 F_7 = F_2 F_8 = F_9 = F_2 F_{10} = F_1 F_{11} = F_{12}.
\]
The matrix $A$ corresponding to these defining relations is given by

$$
A = \begin{bmatrix}
  0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
  1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
  1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
$$

Step (3): The corresponding $C$ matrix is given by

$$
C = AZ = \begin{bmatrix}
  0 & 0 & 1 & 1 \\
  0 & 1 & 0 & 1 \\
  0 & 1 & 0 & 1 \\
  0 & 1 & 0 & 1 \\
  0 & 0 & 1 & 1 \\
  0 & 0 & 1 & 1 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 0 \\
\end{bmatrix}
$$

The parallel-flats design defined by $A$ and $C$ is guaranteed to be an orthogonal design of resolution $(G_1, G_2)$. The combinations of levels of factors $F_{11}$ and $F_{12}$ determine the blocks. We explicitly list the design below.

<table>
<thead>
<tr>
<th>block I</th>
<th>block II</th>
<th>block III</th>
<th>block IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>run 1</td>
<td>0101011101</td>
<td>1101000111</td>
<td>1100110100</td>
</tr>
<tr>
<td>run 2</td>
<td>0000000000</td>
<td>1000110101</td>
<td>1001101001</td>
</tr>
<tr>
<td>run 3</td>
<td>1010101111</td>
<td>0010110101</td>
<td>0011000110</td>
</tr>
<tr>
<td>run 4</td>
<td>1111110100</td>
<td>0111101000</td>
<td>0110011101</td>
</tr>
</tbody>
</table>

Example 4: Consider a $2^{15}$ factorial experiment with

$$
G_1 = \{\mu, F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9, F_{10}, F_{11}, F_{12}, F_{13}, F_{14}, F_{15}, F_1F_2, F_1F_3, F_1F_7, F_1F_8, F_1F_9, F_1F_{10}, F_1F_{11}, F_1F_{13}, F_1F_{14}, F_2F_3, F_2F_4, F_2F_7, F_2F_8, F_2F_{10}, F_2F_{14}, F_1F_2F_{11}, F_2F_7F_3\}
$$

and $G_2$ the empty set. The set $R$ contains 321 effects. Since there are 33 effects to be estimated, the smallest classical fractional factorial orthogonal design for this problem would require at least 64 runs. In fact, a 64-run design can easily be obtained by the Franklin-Bailey algorithm. Here we attempt to find an orthogonal design with $48 = 12 \times 2^{15-13}$ runs.
Step (1): According to our algorithm we must first select a matrix $Z$ of size $15 \times 12$ such that the term in (4.3) is zero for a large number of effects in $R$. We begin with the following $12 \times 12$ Hadamard matrix $H$.

$$H = \begin{bmatrix}
  h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & h_{11} & h_{12} \\
  0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
  1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
  1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
  1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
  0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
  1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
  0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
  1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
  0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
  0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
  1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 
\end{bmatrix}$$

The following table gives the frequency of occurrence of each factor in the effects in $R$ along with the row assignment for each factor using the “cyclic” assignment strategy.

<table>
<thead>
<tr>
<th>Factor</th>
<th>$F_2$</th>
<th>$F_1$</th>
<th>$F_7$</th>
<th>$F_8$</th>
<th>$F_{11}$</th>
<th>$F_{14}$</th>
<th>$F_3$</th>
<th>$F_{10}$</th>
<th>$F_{13}$</th>
<th>$F_9$</th>
<th>$F_5$</th>
<th>$F_{12}$</th>
<th>$F_{15}$</th>
<th>$F_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>143</td>
<td>142</td>
<td>70</td>
<td>70</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>46</td>
<td>44</td>
<td>44</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
</tbody>
</table>

when $i = 0$ $h_2$ $h_3$ $h_4$ $h_5$ $h_6$ $h_7$ $h_8$ $h_9$ $h_{10}$ $h_{11}$ $h_{12}$ $h_2$ $h_3$ $h_4$ $h_5$
when $i = 1$ $h_1$ $h_2$ $h_3$ $h_4$ $h_5$ $h_6$ $h_7$ $h_8$ $h_9$ $h_{10}$ $h_{11}$ $h_{12}$ $h_2$ $h_3$ $h_4$
when $i = 2$ $h_1$ $h_1$ $h_2$ $h_3$ $h_4$ $h_5$ $h_6$ $h_7$ $h_8$ $h_9$ $h_{10}$ $h_{11}$ $h_{12}$ $h_2$ $h_3$
when $i = 3$ $h_1$ $h_1$ $h_1$ $h_2$ $h_3$ $h_4$ $h_5$ $h_6$ $h_7$ $h_8$ $h_9$ $h_{10}$ $h_{11}$ $h_{12}$ $h_2$
when $i = 4$ $h_1$ $h_1$ $h_1$ $h_1$ $h_2$ $h_3$ $h_4$ $h_5$ $h_6$ $h_7$ $h_8$ $h_9$ $h_{10}$ $h_{11}$ $h_{12}$

The choice $i = 2$ leads to a design of the required type. For this choice, the corresponding $Z$ matrix is given by

$$Z = \begin{bmatrix}
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
  1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
  0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
  1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
  1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
  0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
  1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
  1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\
  0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
  0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
  1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 
\end{bmatrix}$$
It can be verified that the set $R_0$ consists of the defining vectors corresponding to the effects

$$
F_1, F_2, F_1F_2, F_1F_2F_3, F_1F_2F_3F_4, F_1F_2F_3F_4F_5, F_1F_2F_3F_4F_5F_6,
$$

$$
F_2F_3F_4F_5F_6, F_2F_3F_4F_5F_6F_7, F_2F_3F_4F_5F_6F_7F_8, F_2F_3F_4F_5F_6F_7F_8F_9,
$$

$$
F_2F_3F_4F_5F_6F_7F_8F_9F_10, F_2F_3F_4F_5F_6F_7F_8F_9F_10F_11, F_2F_3F_4F_5F_6F_7F_8F_9F_10F_11F_12.
$$

Step (2): We use the Franklin-Bailey algorithm with $R_0$ as the set of ineligible effects to find a fractional factorial design in 4 runs for which the expression in (4.4) is zero for every effect in $R_0$. One such solution is given by the aliasing relations

$$
I = F_1F_3 = F_1F_4 = F_5 = F_6 = F_1F_2F_7 = F_1F_2F_8 = F_1F_2F_9 = F_1F_2F_10,
$$

$$
The A matrix corresponding to this set of aliasing relations is given by

$$
A = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

Step (3): The $C$ matrix corresponding to $A$ and $Z$ above is given by

$$
C = AZ = \begin{bmatrix}
1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
$$
The parallel-flats design defined by \( A \) and \( C \) is an orthogonal design of resolution \((G_1, G_2)\) consisting of 48 runs.

To test how successful the algorithm is in obtaining orthogonal parallel-flats designs with small numbers of runs, we created several sets \( G_1 \) of randomly selected primary effects to simulate situations that could occur in practice. \( G_2 \) was taken to be the empty set for this exercise (this is not really critical since the success of the algorithm really only depends on the set \( R \)). Only “nonisomorphic” sets \( G_1 \) were considered; if one random set of primary effects differed from another only by a relabeling of factors, then only one of these was considered (otherwise the algorithm’s success rate would be artificially higher than what is reported below). We considered up to 20 factors and for each case we included all main effects, a specified number of two-factor interactions, and a specified number of three-factor interactions. We constrained the structure of the interactions: each interaction was forced to include at least one of a specified set of 1, 2, 3, or 4 factors. The following table reports the success rate of the algorithm in finding a design smaller than what would be found using the Franklin-Bailey algorithm. In the table, \( n \) is the number of factors, \( k_1 \) is the number of two-factor interactions in \( G_1 \), \( k_2 \) is the number of three-factor interactions in \( G_1 \), \( k_3 \) is the number of “special” factors, at least one of which must appear in every interaction, and \( p \) is the fraction of the total number of cases tried for which an orthogonal design was obtained with number of runs smaller than what would be obtained from the Franklin-Bailey algorithm. Also, \( N \) is the actual number of runs in the design produced by the algorithm. The success rate \( p \) is stated as a fraction \( a/b \) where \( b \) is the total number of “nonisomorphic” effects sets \( G_1 \) tried and \( a \) is the number of times an orthogonal design in \( N \) runs was found with \( N \) smaller than the smallest \( 2^{n-k} \) fractional factorial capable of estimating the effects in \( G_1 \). The value of \( b \) varies depending on \( k_1, k_2, k_3 \) because the random generation of \( G_1 \) was carried out 5000 times or until 100 nonisomorphic \( G_1 \) sets were found, whichever occurred first. For certain values of \( n, k_1, k_2, k_3 \), the total number of nonisomorphic \( G_1 \) sets is easy to calculate (for instance, for \( n = 12, k_1 = 20, k_2 = 0, k_3 = 2 \). it is easy to see that \( b = 1 \), but the calculation is not easily done for most cases.
<table>
<thead>
<tr>
<th>n</th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$k_3$</th>
<th>$p$</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>20</td>
<td>0</td>
<td>2</td>
<td>$1/1$</td>
<td>48</td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>0</td>
<td>3</td>
<td>$100/100$</td>
<td>48</td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>0</td>
<td>4</td>
<td>$1/100$</td>
<td>48</td>
</tr>
<tr>
<td>12</td>
<td>18</td>
<td>2</td>
<td>2</td>
<td>$35/100$</td>
<td>48</td>
</tr>
<tr>
<td>12</td>
<td>18</td>
<td>2</td>
<td>3</td>
<td>$1/100$</td>
<td>48</td>
</tr>
<tr>
<td>12</td>
<td>18</td>
<td>2</td>
<td>4</td>
<td>$0/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>0</td>
<td>2</td>
<td>$48/49$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>0</td>
<td>3</td>
<td>$39/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>0</td>
<td>4</td>
<td>$7/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>1</td>
<td>2</td>
<td>$18/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>1</td>
<td>3</td>
<td>$8/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>1</td>
<td>4</td>
<td>$0/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>14</td>
<td>2</td>
<td>2</td>
<td>$11/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>14</td>
<td>2</td>
<td>3</td>
<td>$2/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>14</td>
<td>2</td>
<td>4</td>
<td>$1/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>20</td>
<td>0</td>
<td>2</td>
<td>$20/28$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>20</td>
<td>0</td>
<td>3</td>
<td>$6/100$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>20</td>
<td>0</td>
<td>4</td>
<td>$0/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>19</td>
<td>1</td>
<td>2</td>
<td>$7/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>19</td>
<td>1</td>
<td>3</td>
<td>$1/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>16</td>
<td>19</td>
<td>1</td>
<td>4</td>
<td>$1/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>0</td>
<td>1</td>
<td>$1/1$</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>0</td>
<td>2</td>
<td>$50/51$</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>0</td>
<td>3</td>
<td>$24/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>0</td>
<td>4</td>
<td>$4/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>0</td>
<td>2</td>
<td>$41/41$</td>
<td>48</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>0</td>
<td>3</td>
<td>$46/100^*$</td>
<td>48</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>0</td>
<td>4</td>
<td>$0/100$</td>
<td>48</td>
</tr>
<tr>
<td>18</td>
<td>46</td>
<td>0</td>
<td>3</td>
<td>$3/3$</td>
<td>80</td>
</tr>
<tr>
<td>18</td>
<td>46</td>
<td>0</td>
<td>4</td>
<td>$26/100$</td>
<td>96</td>
</tr>
</tbody>
</table>

* Not all strategies for choosing $Z$ were used so the numbers reported provide a lower bound on the success rate

The searches that produced 80 and 96 run designs required significant computer time. This generally occurred when the Franklin-Bailey part had no solution, so the program had to perform a complete tree search to conclude that a design consistent with the set $R_0$ did not exist.

Examination of the table indicates that 48-run designs are plentiful in some situations but not others. When a small number of three factor interactions are included, the proportion of times a 48 run design was found is generally smaller than if no three factor interaction is included. In cases where the "success rate" appears to be very low (e.g., 1 out of 100), it may be the case that 48-run designs do not exist for most of the 100 problems tried. The algorithm will find a subset of the ones that actually exist, but the number of cases out of the 100 attempts for which a 48-run design exists is unknown and is generally very difficult to determine. Nevertheless, the results in the table indicate that the algorithm can be expected to be reasonably successful in finding 48-run designs of user-specified resolutions (also some 80-run and some 96-run solutions), and should be useful in practice. Our implementation seems to work satisfactorily for problems involving up to 16 factors. With more factors the cpu time increases rapidly when performing the Franklin-Bailey part of the computations. But it is still feasible to run problems involving up to 20 factors. With further fine tuning of our program we expect to increase this limit.

A copy of the executable version of the program can be obtained from us via email (contact us at hari@stat.colostate.edu). The advantage of our program is that it is capable of finding designs with number of runs not necessarily a power of 2. To our knowledge, none of the commercially available packages currently offer this feature although, as indicated earlier, the FACTEX procedure of SAS/QC® software as well as RS/Discover® software use the Franklin-Bailey (1977) and Franklin (1985) algorithms for finding orthogonal designs with run sizes equal to a power of the appropriate prime. Since our implementation of the
algorithm also has a Franklin-Bailey module as a component, a full Franklin-Bailey search may be made, if the user desires, to find single-flat designs with user-specified resolution (this solution corresponds to selecting $Z$ to be a $N \times 1$ matrix of all zeros). However, a full Franklin-Bailey search can be extremely time consuming when the number of factors is large compared to the number of runs desired.

5 Examples

We now discuss two examples for which designs were generated by our computer program.

**Example 5:** Shoemaker, Tsui and Wu (1991) used the following example to show that the use of combined arrays in place of product arrays (inner array ⊗ outer array) for robust designs can reduce the number of runs. One of the early steps in processing silicon wafers for IC devices is to grow a layer of silicon on polished wafers. There are eight control factors of interest which are $F_1 = $ Rotation method, $F_2 = $ Wafer code, $F_3 = $ Deposition temperature, $F_4 = $ Deposition time, $F_5 = $ Arsenic flow rate, $F_6 = $ HCl etch temperature, $F_7 = $ HCl flow rate and $F_8 = $ Nozzle position each at 2 levels. Location of the wafers (top or bottom) and facet (1, 2, 4, 6) are considered as two noise factors at 2 and 4 levels, respectively. The main effects of the last noise factor can be replaced by 2 main effects and 1 interaction of two pseudo 2-level factors. We rename the noise factors as $F_9, F_{10}, F_{11}$ and $F_{10}F_{11}$. Shoemaker, Tsui and Wu pointed out that a product array for this experiment may need a 128-run design. They consider it reasonable to assume that the interactions

\[
F_3F_9, F_4F_9, F_7F_9, F_8F_9, F_1F_{10}, F_2F_{10}, F_3F_{10}, F_5F_{10}, F_6F_{10}, \\
F_3F_{10}F_{11}, F_4F_{10}F_{11}, F_7F_{10}F_{11}, F_8F_{10}F_{11}
\]  

(5.1)

are negligible, and suggested a 64-run orthogonal design to estimate the nonnegligible effects. By using our algorithm, we find that an orthogonal design with 96 runs in 8 blocks estimates all main effects and all control-by-noise interactions without assuming the effects in (5.1) to be negligible. The design is given below.

<table>
<thead>
<tr>
<th>Block I</th>
<th>Block II</th>
<th>Block III</th>
<th>Block IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>run 1</td>
<td>1111111110</td>
<td>1111111101</td>
<td>1100101010</td>
</tr>
<tr>
<td>run 2</td>
<td>0001001101</td>
<td>0001001101</td>
<td>0010011010</td>
</tr>
<tr>
<td>run 3</td>
<td>1110100011</td>
<td>1111010010</td>
<td>1110100111</td>
</tr>
<tr>
<td>run 4</td>
<td>0001110011</td>
<td>0001100011</td>
<td>0001101000</td>
</tr>
<tr>
<td>run 5</td>
<td>0001101011</td>
<td>1010000001</td>
<td>0010011111</td>
</tr>
<tr>
<td>run 6</td>
<td>1110000100</td>
<td>0100110010</td>
<td>1100110111</td>
</tr>
<tr>
<td>run 7</td>
<td>0000111101</td>
<td>1010011100</td>
<td>0011010001</td>
</tr>
<tr>
<td>run 8</td>
<td>1110011000</td>
<td>0101111111</td>
<td>1101100000</td>
</tr>
<tr>
<td>run 9</td>
<td>0001110010</td>
<td>0001100001</td>
<td>0101000110</td>
</tr>
<tr>
<td>run 10</td>
<td>1110000101</td>
<td>1110000010</td>
<td>1011101011</td>
</tr>
<tr>
<td>run 11</td>
<td>0000111101</td>
<td>0001111110</td>
<td>0100001001</td>
</tr>
<tr>
<td>run 12</td>
<td>1110001100</td>
<td>1110001111</td>
<td>1010110000</td>
</tr>
</tbody>
</table>

20
<table>
<thead>
<tr>
<th>Block V</th>
<th>Block VI</th>
<th>Block VII</th>
<th>Block VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>00010110111</td>
<td>01100100001</td>
<td>01101010010</td>
<td>10010110001</td>
</tr>
<tr>
<td>11101110111</td>
<td>10011110101</td>
<td>10000000101</td>
<td>01110100010</td>
</tr>
<tr>
<td>00011000011</td>
<td>01100010101</td>
<td>01111110101</td>
<td>10000101100</td>
</tr>
<tr>
<td>11110100000</td>
<td>10001011111</td>
<td>10010100000</td>
<td>01101001111</td>
</tr>
<tr>
<td>11000101111</td>
<td>10001010101</td>
<td>10010100000</td>
<td>01101000001</td>
</tr>
<tr>
<td>00101001101</td>
<td>00101001101</td>
<td>01110100100</td>
<td>10000000010</td>
</tr>
<tr>
<td>11010110011</td>
<td>11010110010</td>
<td>11001001100</td>
<td>01111111100</td>
</tr>
<tr>
<td>00111010000</td>
<td>00111010011</td>
<td>01101001000</td>
<td>10010011111</td>
</tr>
<tr>
<td>01100101110</td>
<td>00010110011</td>
<td>10111001110</td>
<td>10111001001</td>
</tr>
<tr>
<td>10011110101</td>
<td>11100111010</td>
<td>01010101101</td>
<td>01010101010</td>
</tr>
<tr>
<td>01100010111</td>
<td>00110010010</td>
<td>10101010010</td>
<td>10101010100</td>
</tr>
<tr>
<td>10001101000</td>
<td>11110100111</td>
<td>01000110000</td>
<td>01000110111</td>
</tr>
</tbody>
</table>

This block design may be a more practical design to implement, especially if the number of runs that can be performed within a given shift (or day) is restricted.

**Example 6:** For a robust process design, suppose \( F_1, F_2, \ldots, F_9, F_{10} \) are control factors and \( F_{11}, F_{12} \) are noise factors, each with 2-levels. The experimenter is interested in estimating all main effects and all control-by-noise interactions. We look for an orthogonal design with 48 runs in 4 blocks (each with 12 runs). Using our algorithm we obtain the following design.

<table>
<thead>
<tr>
<th>Block I</th>
<th>Block II</th>
<th>Block III</th>
<th>Block IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000001</td>
<td>00000000001</td>
<td>11000011101</td>
<td>11000011101</td>
</tr>
<tr>
<td>00000000000</td>
<td>00000000000</td>
<td>11000011100</td>
<td>11000011100</td>
</tr>
<tr>
<td>10011001111</td>
<td>10011001111</td>
<td>00101101110</td>
<td>00101101110</td>
</tr>
<tr>
<td>10011001110</td>
<td>10011001110</td>
<td>00101101100</td>
<td>00101101100</td>
</tr>
<tr>
<td>01101010111</td>
<td>01101010111</td>
<td>11001100011</td>
<td>11001100011</td>
</tr>
<tr>
<td>01101010110</td>
<td>01101010110</td>
<td>11001100010</td>
<td>11001100010</td>
</tr>
<tr>
<td>11110100101</td>
<td>11110100101</td>
<td>00010110111</td>
<td>00010110111</td>
</tr>
<tr>
<td>11110100100</td>
<td>11110100100</td>
<td>00010110100</td>
<td>00010110100</td>
</tr>
<tr>
<td>10100111011</td>
<td>10100111011</td>
<td>01110001011</td>
<td>01110001011</td>
</tr>
<tr>
<td>10100111010</td>
<td>10100111010</td>
<td>01110001000</td>
<td>01110001000</td>
</tr>
<tr>
<td>01011111000</td>
<td>01011111000</td>
<td>10111010001</td>
<td>10111010001</td>
</tr>
<tr>
<td>01011111000</td>
<td>01011111000</td>
<td>10111010000</td>
<td>10111010000</td>
</tr>
</tbody>
</table>

# 6 Concluding Remarks

We have provided an algorithm that can be useful for constructing "small" fractional factorial designs to suit the needs of the user who specifies the set of primary effects \( G_1 \) and the set of secondary effects \( G_2 \). A design produced by our algorithm may at times turn out to be a previously known or tabulated design. However, the direct use of a tabulated design for a given problem will require proper assignment of factor labels to columns of the design to
ensure that the effects of interest can be estimated. This will often require trial-and-error and/or ingenuity and is not at all automatic except in small problems. Automating this trial-and-error approach does not appear to be as practical as a direct construction method such as the Franklin-Bailey algorithm for single-flat designs or our algorithm for the wider class of parallel-flats designs.

A computer program based on this algorithm has been implemented and is available from the authors (email contact: hari@stat.colostate.edu). In tests carried out to evaluate the success rate of the program, it found orthogonal designs with run sizes smaller than what would be required by traditional designs (48 runs instead of 64; 80 or 96 runs instead of 128) sufficiently often that we feel it may be useful to practitioners. We are currently investigating ways to improve the algorithm and the software to make its use practical even for problems that are larger than what we have reported here.

Acknowledgements: We wish to thank Dr. Li and Professor J. N. Srivastava for helpful discussions. Thanks are also due to three referees and an associate editor who made many significant suggestions which helped improve the paper both in style and in content.

SAS® and SAS/QC® are registered trademarks of SAS Institute Inc. in the USA and other countries. RS/Discover® is a registered trademark of BBN Inc.

References


