Optimal Projective Three-Level Designs for Factor Screening and Interaction Detection

Hongquan Xu
Department of Statistics
University of California
Los Angeles, CA 90095-1554
(hqxu@stat.ucla.edu)

Shao-Wei Cheng
Institute of Statistical Science
Academia Sinica
Taipei 115, Taiwan
(swcheng@stat.sinica.edu.tw)

C. F. J. Wu
Department of Statistics
University of Michigan
Ann Arbor, MI 48109-1092
(jeffwu@umich.edu)

Orthogonal arrays are widely used in industrial experiments for factor screening. Suppose only a few of the factors are important. An orthogonal array can be used not only for screening factors but also for detecting interactions among a subset of active factors. In this article, a set of optimality criteria is proposed to assess the performance of designs for factor screening, projection, and interaction detection, and a three-step approach is proposed to search for optimal designs. Combinatorial and algorithmic construction methods are proposed for generating new designs. Level permutation methods are used for improving the eligibility and estimation efficiency of the projected designs. The techniques are then applied to search for best three-level designs with 18 and 27 runs. Many new, efficient and practically useful nonregular designs are found and their properties discussed.

KEY WORDS: Contamination; Factor sparsity; Generalized minimum aberration; Orthogonal array; Projection aberration; Projection-efficiency criteria.

1 Introduction

For run size economy, *orthogonal arrays* (OA) are widely used in industrial experiments to screen important factors from a large number of potential factors. Traditionally, design and analysis of screening experiments has been restricted to main effects only by assuming that the interactions are negligible. Hamada and Wu (1992) went beyond the traditional approach and proposed an analysis strategy to demonstrate that some interactions could be identified beyond a few significant main effects.

For illustration, consider an experiment reported by King and Allen (1987) that studied axial lead 4.7 microhenry radio frequency chokes. A radio frequency choke is a circuit element designed to present a high impedance to radio frequency energy while offering minimal resistance to direct current. The objective of the experiment was to identify important factors and best settings in the choke winding operation. In the experiment, an 18-run OA was used to study one two-level factor (A) and seven three-level factors (B–H), and each run had two replicates. The response was a 10-piece sampling of the self resonating frequency in Mhz. The design matrix and the responses are given in Table 1.

Motivated by the procedure of Hamada and Wu (1992), we perform a two-stage analysis on this experiment. At the first stage, we fit an analysis of variance (ANOVA) model for main effects and find that four factors B, E, G, and H are significant. At the second stage, we consider models that consist of the main effects of the four significant factors and some of two-factor interactions among them. Since there are six two-factor interactions, each having four degrees of freedom, the 18-run experiment does not have enough degrees of freedom to estimate all two-factor interactions among them. One possible strategy is to use an iterative stepwise regression procedure to identify significant effects as in Hamada and Wu (1992). Since all factors are quantitative, we take an alternative approach and fit a second-order model among the four active factors, which has 15 unknown parameters. We find that all linear effects x_B, x_E, x_G, and x_H, and five quadratic effects x_E^2, x_Bx_E, x_Ex_G, x_Ex_H, and x_Gx_H are significant (at the 5% level). The R^2 of this model is 0.96, indicating that the model fits the data well. Notice that the identification of the four significant interactions is achieved through the projection of the design matrix onto active factors, which serves as a link between screening a larger number of factors and the more intensive study of the response surface over a smaller number of important factors.

The previous two-stage analysis was first suggested by Cheng and Wu (2001, henceforth abbre-

Table 1: Design Matrix and Responses, Radio Frequency Chokes Experiment

Run	A	B	C	D	E	F	G	H	Responses	
1	0	0	0	0	0	0	0	0	106.20	107.70
2	0	0	1	1	1	1	1	1	104.20	102.35
3	0	0	2	2	2	2	2	2	85.90	85.90
4	0	1	0	0	1	1	2	2	101.15	104.96
5	0	1	1	1	2	2	0	0	109.92	110.47
6	0	1	2	2	0	0	1	1	108.91	108.91
7	0	2	0	1	0	2	1	2	109.76	112.66
8	0	2	1	2	1	0	2	0	97.20	94.51
9	0	2	2	0	2	1	0	1	112.77	113.03
10	1	0	0	2	2	1	1	0	93.15	92.83
11	1	0	1	0	0	2	2	1	97.25	100.6
12	1	0	2	1	1	0	0	2	109.51	113.28
13	1	1	0	1	2	0	2	1	85.63	86.91
14	1	1	1	2	0	1	0	2	113.17	113.45
15	1	1	2	0	1	2	1	0	104.85	98.87
16	1	2	0	2	1	2	0	1	113.14	113.78
17	1	2	1	0	2	0	1	2	103.19	106.46
18	1	2	2	1	0	1	2	0	95.70	97.93

viated CW). Formally, the two-stage analysis is as follows:

Stage 1. Perform ANOVA for factor screening and identify important factors.

Stage 2. Fit a second-order model to the factors identified in Stage 1.

This analysis strategy assumes that only a few factors are active in a factorial experiment, called *factor sparsity* by Box and Meyer (1986), and that significant interactions appear only among these active factors, called strong *effect heredity* by Chipman (1996) and functional marginality by McCullagh and Nelder (1989, chap. 3). These are empirical principles whose validity has been confirmed in many real experiments. The assumption of strong effect heredity can be restrictive in some cases. If a factor's significance is manifested through its interactions with other factors, but not through its main effects, it may be missed in the first stage analysis. To circumvent this problem, one can use a more elaborate procedure like Bayesian methods for factor screening in the first stage. See Box and Meyer (1993), Chipman, Hamada, and Wu (1997), and Wu and Hamada (2000, chap. 8) for further discussions and examples.

Standard *response surface methodology* has three stages: an initial factor screening stage, a stage of sequential experimentation to determine the region of an optimum, and a final stage involving the fitting of a second order model in this region to understand the nature of the optimum. Typically *separate* experiments and designs are used for different stages. However, it is sometimes difficult or impossible to perform the experiments sequentially (see Steinberg and Bursztyn (2001) for an example). It is thus desirable to have a methodology that allows factor screening and response surface exploration to be conducted on the *same* experiment using *one* design. CW argued that the two-stage analysis is a useful alternative to the standard response surface methodology if the design region is appropriate for studying second-order curvatures. Here we should point out that the region does *not* need to contain the optimum. As long as it contains a curved part of the surface, we can study a second-order surface. That is, a second-order experiment can be performed before reaching the optimum. Details on response surface methodology can be found in texts like Box and Draper (1987), Myers and Montgomery (1995), and Khuri and Cornell (1996).

This paper considers the design problem associated with the previous two-stage analysis. Assume that only a few of the factors are identified as significant by performing ANOVA for main effects. We can use an OA for factor screening (estimating main effects, including curvature effects) and by projections we can also study interactions (in our formulation, the linear-by-linear interactions) for a subset of active factors. Since experimenters do not know in advance which factors are important and what the final model will be, it is important to choose a screening design that can entertain as many models as possible. Note that the standard optimum design approach does not apply here since the number of runs is not enough to fit a second-order model at the screening stage. Tsai, Gilmour, and Mead (2000) considered a similar problem and studied the projective properties of three-level designs with 18 runs. The approach we take here is quite different from theirs.

In Section 2, we review the generalized minimum aberration criterion (Xu and Wu 2001) for factor screening and the projection-efficiency criteria (Cheng and Wu 2001) for interaction detection. Then we propose a new criterion to combine these two objectives and a *three-step approach for design search.* Section 3 considers the construction methods of OAs. In order to construct new and efficient designs, we propose a combinatorial method as an extension of Wang and Wu (1991), an algorithmic search due to Xu (2002), and two versions of a search algorithm for level permutations. The techniques are then applied to search for best three-level designs with 18 and 27 runs. Many new, efficient and practically useful nonregular designs are found and their properties discussed in Section 4. Discussion and further remarks are given in Section 5.

2 Optimality Criteria

2.1 The ANOVA Model and Generalized Minimum Aberration Criterion

For factor screening, we adopt the *generalized minimum aberration* (GMA) criterion (Xu and Wu 2001), which is an extension of the minimum aberration criterion (Fries and Hunter 1980).

For a factorial design with N runs and n factors, the (full) ANOVA model is

$$Y = X_0\alpha_0 + X_1\alpha_1 + \cdots + X_n\alpha_n + \varepsilon, \tag{1}$$

where Y is the vector of N observations, α_0 the general mean, α_1 the vector of main effects, α_j the vector of j-factor interactions, X_0 the vector of 1's, X_j the matrix of contrast coefficients for α_j, and ε the vector of independent random errors. Here we consider only the cases where the contrast coefficient of an interaction effect is the product of its corresponding contrast coefficients of main effects. For a two-level factor, the contrast vector of a main effect is $(-1, 1)$; for a three-level factor, the contrast vectors of the linear and quadratic main effects are $(-\sqrt{3/2}, 0, \sqrt{3/2})$ and $(1/\sqrt{2}, -\sqrt{2}, 1/\sqrt{2})$, respectively. For $j = 0, 1, \ldots, n$, Xu and Wu (2001) defined A_j, a function of X_j, to measure the overall aliasing (or correlation) between all j-factor interactions and the general mean. Specifically, if $X_j = [x_{ik}^{(j)}]$, let

$$A_j = N^{-2} \sum_k \left| \sum_{i=1}^N x_{ik}^{(j)} \right|^2 . \tag{2}$$

The vector $(A_1, \ldots, A_n)$ is called the *generalized wordlength pattern* (since A_j is the number of words of length j for a two-level regular design). Xu and Wu (2001) showed that for an OA (of strength 2), $A_1 = A_2 = 0$. The GMA criterion is to sequentially minimize $A_1, A_2, A_3, \ldots$.

Two designs are called *combinatorially isomorphic* if the design matrix of one design can be obtained from that of the other by permutations of rows, columns and levels in the columns. Because combinatorially isomorphic designs have the same generalized wordlength pattern, they are indistinguishable under the GMA criterion.

Note that the computation of A_j according to (2) is cumbersome because it involves all possible projections onto j factors. Alternative efficient computations can be found in Xu and Wu (2001) and Xu (2003).

Example 1. Consider choosing six columns from the commonly used $OA(18, 3^7)$ given in Table 10(i). There are seven possible choices. For illustration, consider three choices. Let $\mathbf{d}_1$, $\mathbf{d}_2$ and $\mathbf{d}_3$ be the resulting design from omitting the first, second and third column, respectively. The generalized wordlength patterns for the three designs are $(0, 0, 10, 22.5, 0, 7)$, $(0, 0, 13, 13.5, 9, 4)$, and $(0, 0, 13, 13.5, 9, 4)$, respectively. Hence, $\mathbf{d}_1$ is the best according to the GMA criterion. Note that $\mathbf{d}_2$ and $\mathbf{d}_3$ have the same generalized wordlength pattern and therefore the GMA criterion cannot distinguish between them.

To explain why GMA is suitable for screening out poor designs, consider the estimation of main effects in the presence of two-factor interactions. Specifically, assume that three and higher-order interactions are negligible. Then the ANOVA model (1) becomes

$$Y = X_0\alpha_0 + X_1\alpha_1 + X_2\alpha_2 + \varepsilon, \tag{3}$$

which contains the constant effect, main effects, and two-factor interactions. At the screening stage, it is common that the degrees of freedom are not enough to fit model (3); therefore, a main effects model, i.e., with $\alpha_2 = 0$ in (3), is fitted instead. For a balanced design (i.e., all levels occur equally often for each factor), an unbiased estimate of the main effects α_1 is $\hat{\alpha}_1 = (X_1'X_1)^{-1}X_1'Y$. However, under the true model (3),

$$E(\hat{\alpha}_1) = \alpha_1 + C\alpha_2,$$

where $C = (X_1'X_1)^{-1}X_1'X_2$ is the alias matrix (see Box and Draper 1987, Section 3.10; Wu and Hamada 2000, Section 8.1 for details). In other words, the estimation of the main effects are contaminated by non-negligible two-factor interactions (Box and Draper 1987, p. 67). A good design should have a small contamination. This leads to the *minimum contamination* criterion, that is, the minimization of $\|C\|^2 = \sum |c_{ij}|^2$ if $C = (c_{ij})$ (Tang and Deng 1999; Xu and Wu 2001; Steinberg and Bursztyn 2001). Xu and Wu (2001) showed that the contamination $\|C\|^2$ is related to the A_3 value for an OA:

$$\|C\|^2 = 3A_3. \tag{4}$$

Recall that A_3 measures the overall aliasing between all three-factor interactions and the general mean. Equation (4) holds because a three-factor interaction is the product of a main effect and a two-factor interaction. For an OA with smaller A_3, its main effects suffer less contamination when a main effects model is fitted, and therefore, factor screening is more effective. For this reason, we use the *minimization of A_3 as the criterion for factor screening.*

2.2 The Second-Order Model and Projection-Efficiency Criteria

For interaction detection, we consider a second-order model and adopt the *projection-efficiency* criteria (Cheng and Wu 2001).

Assume all factors are quantitative and denoted by $x_1, x_2, \ldots, x_n$. Then the second-order model for these factors is given by

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \beta_{ii} x_i^2 + \sum_{1=i<j}^{n} \beta_{ij} x_i x_j + \epsilon, \tag{5}$$

where ϵ is the error term.

Note that the second-order model (5) is different from the ANOVA model (3). For a three-level design, a two-factor interaction has four orthogonal components: linear-by-linear, linear-by-quadratic, quadratic-by-linear, and quadratic-by-quadratic, each having one degree of freedom. The second-order model includes only the linear-by-linear component of a two-factor interaction while the ANONA model includes all four components. For n factors with three levels, the second-order model (5) has $(n+1)(n+2)/2$ parameters while the ANOVA model (3) has $2n^2+1$ parameters. The main reason for considering the second-order model for interaction detection is that in many cases the degrees of freedom are not enough to entertain model (3) but may be enough to entertain the second-order model (5). For example, for three factors (with three levels), an 18-run design can entertain model (5) but not model (3); for four factors, a 27-run design can entertain model (5) but not model (3).

Since we do not know in advance which of the components are significant, considering all components in the screening stage seems prudent. A design that does not perform well for model (3) is unlikely to do well for model (5). Therefore, the GMA criterion can efficiently screen out designs that are not suitable for the dual purposes of factor screening and interaction detection.

Some definitions are now in order. A design for n factors is called a *second-order design* if all the parameters in model (5) are estimable. A *projected design* is said to be *eligible* if it is a second-order design; otherwise, it is said to be *ineligible*. A design is called *regular* if it can be constructed through the defining contrast subgroup among its factors; otherwise, it is called *nonregular*. The 2^{n-k} and 3^{n-k} series of designs are regular designs and many mixed-level OAs are nonregular. Details on these concepts and results can be found in Wu and Hamada (2000).

The *projection-efficiency* criteria proposed by CW are as follows:

(i) The number of eligible projected designs should be large, and lower-dimensional projections are more important than higher-dimensional projections;

(ii) Among the eligible projected designs the estimation efficiency as measured by some optimality criterion should be high.

They studied three-level designs with 18, 27 and 36 runs using these criteria. A major finding is that nonregular designs are more efficient than regular designs. Apart from studying the three classes of designs, they did not examine the important issue of choosing optimal designs for the dual purposes of factor screening and interaction detection.

For a design of size N, let X be the model matrix of (5) and $M = X^{'}X/N$ be the moment matrix. A *D-optimal* design maximizes $|M|$, the determinant of M. Kiefer (1961) and Farrell, Kiefer, and Walbran (1967) showed that the D-optimal *continuous* design for model (5) is supported on a subset of points of the 3^n factorial. Let $\mathbf{d}^*$ be the D-optimal continuous design, i.e., $|M(\mathbf{d}^*)| = \max_{\mathbf{d}} |M(\mathbf{d})|$. Then the D-efficiency of a design $\mathbf{d}$ is defined to be

$$D_{\text{eff}} = (|M(\mathbf{d})|/|M(\mathbf{d}^*)|)^{1/p}, \tag{6}$$

where $p = (n+1)(n+2)/2$ is the number of parameters in model (5).

Here we quantify the definition of the projection-efficiency criteria in order to rank designs. Let E_i denote the number of eligible i-factor projections and $\bar{D}_i$ the average D-efficiency of all eligible

Table 2: Eligible Projections, Estimation Efficiency, Overall A_3 Values, and Projection Frequencies

Design	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$	Overall A_3	Proj. A_3 Freq. $\frac{1}{2}$	$\frac{2}{3}$	1	2
$\mathbf{d}_1$	**20**	**15**	0	0.89	0.74	0	10	20	0	0	0
$\mathbf{d}_2$	19	12	0	0.88	0.7	0	13	16	0	3	1
$\mathbf{d}_3$	**20**	**15**	0	0.87	0.69	0	13	14	0	6	0

i-factor projections, where the D-efficiency is calculated as in (6). Then the projection-efficiency criteria can be restated as:

(i) To sequentially maximize the eligibility $E_3, E_4, E_5, \ldots$;

(ii) Among those designs with the same eligibility, to sequentially maximize the average D-efficiency of eligible projections $\bar{D}_3, \bar{D}_4, \bar{D}_5, \ldots$.

Example 2. (Continued from Example 1) For each design, there are 20 three-factor projections, 15 four-factor projections and six five-factor projections. Table 2 lists the number of eligible projections E_i, estimation efficiency $\bar{D}_i$, and some other properties to be explained later. Since the second-order model has 21 parameters for five factors, any five-factor projection of an 18-run design is ineligible and hence $E_5 = 0$. For $\mathbf{d}_1$ and $\mathbf{d}_3$, all 20 three-factor projections and 15 four-factor projections are eligible (i.e., $E_3 = 20$ and $E_4 = 15$). (Whenever this happens, boldface is used for the eligible numbers in the tables.) For $\mathbf{d}_2$, one three-factor projection and three four-factor projections are ineligible (i.e., $E_3 = 19$ and $E_4 = 12$). Furthermore, $\mathbf{d}_1$ is better than $\mathbf{d}_3$ in terms of $\bar{D}_3$ and $\bar{D}_4$. In summary, $\mathbf{d}_1$ is the best and $\mathbf{d}_2$ is the worst under the projection-efficiency criteria.

The projection-efficiency criteria have one major shortcoming: They are computationally intensive because the computation of D-efficiency for *all* possible projections is required.

2.3 Projection Aberration Criterion

Here we propose a new criterion to combine factor screening and interaction detection.

When a design with n factors is projected onto any three factors, it produces $\binom{n}{3}$ three-factor projected designs. Each of these designs has an A_3 value, which is referred to as the *projected* A_3 value. We use *overall* A_3 to denote the A_3 value calculated from the whole n-factor design.

The frequency of the projected A_3 values is called *projection frequency*. Since large projected A_3 values are deemed undesirable, we propose the *projection aberration criterion* which sequentially minimizes the projection frequency starting from the largest projected A_3 value.

A lemma that explains the relationship among overall A_3, projected A_3, and projection frequency follows:

Lemma 1. *The overall A_3 value of a design equals the sum of all its projected A_3 values, i.e., the sum of distinctive projected A_3 values multiplied by their corresponding frequencies.*

The proof is straightforward and omitted. The lemma suggests that the projected A_3 values and projection frequency present more detailed information than the overall A_3. It is reasonable to expect that a design with a small overall A_3 value will have low projection aberration. For two-level designs, Deng and Tang (2002) showed that the contamination criterion and the projection aberration criterion are generally consistent in ranking designs. The situation is more complicated for three-level designs. As will be seen in Section 4, the two criteria are consistent for ranking 18-run designs but not for 27-run designs.

Example 3. (Continued from Example 1) For each design, there are 20 three-factor projections, each having a projected A_3 value. The frequencies of projected A_3 values are listed in Table 2. Among the three designs, $\mathbf{d}_2$ is the worst under the projection aberration criterion since one of its 3-factor projections has projected $A_3 = 2$; $\mathbf{d}_1$ is again the best because all its 3-factor projections have projected $A_3 = 0.5$. The projection aberration criterion and the projection-efficiency criteria produce the same ranking for the three designs while the GMA criterion cannot distinguish between $\mathbf{d}_2$ and $\mathbf{d}_3$.

There is a close connection between projected A_3 values and eligibility. A key finding in CW is that the presence of projections with three-letter words causes low projection-efficiency. This is referred to as *curse of three-letter words*. The main reason behind the curse is insufficient degrees of freedom for fitting a second-order model, which has 10 parameters for three factors. Three columns form a three-letter word if the level combinations of any two columns completely determine the level of the third column. For example, columns 1, 3 and 4 of Table 10(i) form a three-letter word. It is clear that a three-factor projection (with three levels) has only nine distinct runs if the three factors form a three-letter word. The following lemma shows the relationship between projected A_3 values and curse of three-letter word, and gives a necessary and sufficient condition for a three-factor projection to be free of three-letter words.

Lemma 2. *For an s-level OA, the projected A_3 value of any three-factor projection is less than or equal to $s-1$, and the equality holds if and only if the three factors form a three-letter word.*

Its proof is given in the Appendix. From Lemma 2, a three-factor projection of a three-level design is free of three-letter word if and only if its projected A_3 value is strictly less than two. Since the projection aberration criterion first minimizes the frequency of projected A_3 value of 2, optimal designs based on it would have a maximum number of eligible three-factor projections.

CW showed that for regular designs, any three-factor or four-factor projection is eligible if it is free of any three-letter word. Our study for 18- and 27-run designs suggests that the following may be true for general designs: "any three-factor projection is eligible if it is free of any three-letter word". We have counter examples (see Section 4.1) to show that a four-factor projection can be ineligible even if it is free of any three-letter word.

There is also a close relationship between projected A_3 values and estimation efficiency. The projected A_3 value captures an important property of three-factor projection, i.e., the number of distinct runs.

Example 4. Consider 3-factor projections from 18-run OAs. A complete search shows that there are four combinatorially-nonisomorphic 3-factor projections, which can be obtained by choosing three columns of the OA in Table 10(ii). Their A_3 values, numbers of distinct runs and average D_3 values are as follows:

A_3	$\frac{1}{2}$	$\frac{2}{3}$	1	2
Distinct Runs	18	17	15	9
Average D_3	0.882	0.864	0.82	0

where the average D_3 is the average D-efficiency of all possible 27 combinatorially-isomorphic designs resulted from level permutations (see Section 3.1 for discussion on level permutations). Evidently the A_3 value completely characterizes the estimation efficiency in this example.

Example 5. Consider 3-factor projections from 27-run OAs. A complete search shows that there are nine combinatorially-nonisomorphic 3-factor projections. Their A_3 values, numbers of distinct runs and average D_3 values are as follows:

A_3	0	$\frac{8}{27}$	$\frac{4}{9}$	$\frac{14}{27}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{20}{27}$	$\frac{10}{9}$	2
Distinct Runs	27	23	21	20	19	18	18	15	9
Average D_3	0.932	0.904	0.889	0.881	0.864	0.864	0.855	0.805	0

A 3^3 full factorial design has 27 distinct runs and $A_3 = 0$ while a regular 3^{3-1} design has 9 distinct runs and $A_3 = 2$. A nonregular $OA(27, 3^3)$ has an A_3 value between 0 and 2. Again, it is evident that the smaller the A_3 value, the larger the number of distinct runs, and hence the higher the estimation efficiency.

2.4 A Three-Step Approach for Design Search

CW noted that combinatorial isomorphism cannot fully discriminate between designs. Through level permutation, a design can be changed to combinatorially isomorphic designs with different geometric structures and thus different eligibility and estimation efficiency. They referred to these designs as *model nonisomorphic* designs. Because combinatorially isomorphic designs have the same generalized wordlength pattern, neither GMA nor projection aberration criterion can discriminate between them. After optimal designs are chosen based on these criteria, the levels of factors of the design should be permuted to further improve eligibility and estimation efficiency. Summarizing the discussions in this section, we propose the following three-step approach to search for optimal projective designs for factor screening and interaction detection among OAs.

Step 1. Use the overall A_3 value to screen out poor OAs for factor screening.

Step 2. Apply the projection aberration criterion to select a best design among the designs chosen in Step 1.

Step 3. Determine the best level permutations of the design chosen in Step 2 for further improvement on eligibility and estimation efficiency.

In the three-step approach, various design properties are sequentially examined. For each step, only those designs that are qualified in the previous step are kept for later comparison. As mentioned earlier, the overall A_3 value and the projection aberration criterion may not be consistent in ranking designs. Designs with large overall A_3 values but low projection aberration may be screened out in Step 1 and therefore have no chance to be compared in Step 2. When the computational load is not too heavy or the number of designs for comparison not too large, it is recommended that both criteria should be applied to *all* designs. When the two criteria lead to different rankings, some trade-off is required. When the number of factors is large, factor screening is probably more important than interaction detection; hence, a design with minimum overall A_3 value is preferred.

On the other hand, for a small to moderate number of factors, interaction detection is probably more important, which favors designs with less projection aberration.

3 Construction Methods for Nonregular Orthogonal Arrays

Traditionally, only a few OAs are used for a given run size and many commonly used OAs are regular designs. As CW observed, regular designs are not efficient for the dual purposes. Here we consider methods for constructing nonregular OAs. Section 3.1 presents a combinatorial method and level permutations that can produce a large number of nonisomorphic OAs. For a systematic study, Section 3.2 proposes computational algorithms to handle the huge collection of designs.

3.1 A Combinatorial Method and Level Permutations

Here we present a general construction method, which is an extension of the difference matrix method in Wang and Wu (1991). Let G be an additive group of s elements denoted by $\{0, 1, \ldots, s-1\}$, s being a prime number. An $n \times k$ matrix with elements from G, denoted by $D_{n,k;s}$, is called a *difference matrix* if, among the differences, modulus s, of the corresponding elements of any two columns, each element of G occurs exactly n/s times. For two matrices U of order $n_1 \times m_1$ and $V = (v_{ij})$ of order $n_2 \times m_2$, define their *Kronecker sum* to be $U \oplus V = (U^{v_{ij}})$, a matrix of order $(n_1 n_2) \times (m_1 m_2)$, where each partition $U^{v_{ij}} = (U + v_{ij}) \pmod s$ is a matrix of order $n_1 \times m_1$. It is known (e.g., Wang and Wu 1991) that the Kronecker sum $U \oplus V$ is an $OA(NM, s^{lk})$ if U is an $OA(N, s^l)$ and V is a difference matrix $D_{M,k;s}$. Here we generalize the Kronecker sum as follows. For two $n \times m$ partitioned matrices $U = [U_{ij}]$ and $V = [V_{ij}]$, define the *generalized Kronecker sum* to be $U \otimes V = [U_{ij} \oplus V_{ij}]$. Let

$$U_i = (U_{i1}, \ldots, U_{im}) \text{ and } V_j = \begin{pmatrix} V_{1j} \\ \vdots \\ V_{nj} \end{pmatrix}.$$

Suppose U_{ij} is an $OA(N, s_j^{l_j})$, U_i is an $OA(N, s_1^{l_1} \cdots s_m^{l_m})$ and V_j is a difference matrix $D_{M,k_j;s_j}$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m$. It can be shown, by following the proof in Wang and Wu (1991), that $U \otimes V$ is an $OA(NM, s_1^{l_1 k_1} \cdots s_m^{l_m k_m})$. It is important to note that both row and column partitions are allowed for the matrix U while in Wang and Wu's original construction, only column partitions are allowed for the matrix U. Therefore, their method is a special case with $n = 1$. Following Wang and Wu (1991), we can enlarge the OAs by adding $0_N \oplus L_M$ to $U \otimes V$, where 0_N is the

$N \times 1$ vector of zeros and L_M is an OA of M runs. Many combinatorially-nonisomorphic OAs can be constructed by choosing various difference matrices for each column of the OAs or by choosing different OAs for each row of the difference matrix.

This generalization allows the construction of nonregular OAs such as $OA(27, 3^{13})$ in Table 11(i). The key step is to choose a different OA for each row of the difference matrix. This is illustrated in the following example.

Example 6. Let $a = (0,0,0,1,1,1,2,2,2)'$, $b = (0,1,2,0,1,2,0,1,2)'$, $c = a + b$, $d = a - b$, $U_1 = (a, b, c, d)$, $U_2 = (a, b, 2c+2, d+1)$, $V_1 = (0,0,0)$, $V_2 = (0,1,2)$, and $V_3 = (0,2,1)$, modulus 3. Then

$$\left[\begin{pmatrix} U_1 \\ U_1 \\ U_1 \end{pmatrix} \otimes \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix}, 0_9 \oplus L_3\right] = \begin{bmatrix} U_1 \oplus V_1 & 0_9 \\ U_1 \oplus V_2 & 1_9 \\ U_1 \oplus V_3 & 2_9 \end{bmatrix}$$

and

$$\left[\begin{pmatrix} U_1 \\ U_2 \\ U_1 \end{pmatrix} \otimes \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix}, 0_9 \oplus L_3\right] = \begin{bmatrix} U_1 \oplus V_1 & 0_9 \\ U_2 \oplus V_2 & 1_9 \\ U_1 \oplus V_3 & 2_9 \end{bmatrix}$$

are two nonisomorphic $OA(27, 3^{13})$'s, where $L_3 = (0, 1, 2)'$ and i_9 is the 9×1 vector of i's. It is more convenient to rewrite these two designs in the following form:

$$\begin{pmatrix} a & b & c & d & a & b & c & d & a & b & c & d & 0_9 \\ a & b & c & d & a+1 & b+1 & c+1 & d+1 & a+2 & b+2 & c+2 & d+2 & 1_9 \\ a & b & c & d & a+2 & b+2 & c+2 & d+2 & a+1 & b+1 & c+1 & d+1 & 2_9 \end{pmatrix} \pmod 3,$$

and

$$\begin{pmatrix} a & b & c & d & a & b & c & d & a & b & c & d & 0_9 \\ a & b & 2c+2 & d+1 & a+1 & b+1 & 2c & d+2 & a+2 & b+2 & 2c+1 & d & 1_9 \\ a & b & c & d & a+2 & b+2 & c+2 & d+2 & a+1 & b+1 & c+1 & d+1 & 2_9 \end{pmatrix} \pmod 3.$$

It is easy to see that the first design is isomorphic to a regular design while the second design, given in Table 11(i), is not.

When level permutations are also considered, more model-nonisomorphic OAs can be constructed. For simplicity, only three-level designs are considered here. The extension to general levels is obvious. Table 3 shows six different level permutations for a three-level factor, which can also be expressed through modulo operation: $p_0(x) = x$, $p_1(x) = x + 1$, $p_2(x) = x + 2$, $p_3(x) = 2x$, $p_4(x) = 2x + 1$, and $p_5(x) = 2x + 2 \pmod 3$ for $x = 0, 1, 2$. It is easy to see from Table 3 that p_5, p_4, and p_3 are reflections of p_0, p_1, and p_2, respectively. That is, p_5 (resp. p_4 and p_3) becomes

Table 3: Six Level Permutations for a Three-Level Factor

	Permutations					
Level	p_0	p_1	p_2	p_3	p_4	p_5
0	0	1	2	0	1	2
1	1	2	0	2	0	1
2	2	0	1	1	2	0

p_0 (resp. p_1 and p_2) if levels 0 and 2 are exchanged. Because reflection of a factor (around level 1) does not change the geometric structure of a design, the projection properties, such as E_i's and $\bar{D}_i$'s, are unchanged under reflection. Therefore, only p_0, p_1, and p_2 shall be considered among the six permutations.

Example 7. In the set-up for Example 6, consider an $OA(27, 3^8)$:

$$\mathbf{d}_4 : \begin{pmatrix} a & b & c & d & a & c & b & d \\ a & b & 2c+2 & d+1 & a+1 & 2c & b+2 & d \\ a & b & c & d & a+2 & c+2 & b+1 & d+1 \end{pmatrix} \pmod 3,$$

which is a subdesign (columns 1-5,7,10,12) of Table 11(i). Applying p_3 on the 6th and 8th columns, we obtain

$$\mathbf{d}_5 : \begin{pmatrix} a & b & c & d & a & 2c & b & 2d \\ a & b & 2c+2 & d+1 & a+1 & c & b+2 & 2d \\ a & b & c & d & a+2 & 2c+1 & b+1 & 2d+2 \end{pmatrix} \pmod 3,$$

which is the nonregular design constructed by CW via a different (but related) method. Applying p_1 on the 6th and 8th columns and p_2 on the 7th column of $\mathbf{d}_4$, we obtain

$$\mathbf{d}_6 : \begin{pmatrix} a & b & c & d & a & c+1 & b+2 & d+1 \\ a & b & 2c+2 & d+1 & a+1 & 2c+1 & b+1 & d+1 \\ a & b & c & d & a+2 & c & b & d+2 \end{pmatrix} \pmod 3.$$

These three designs are combinatorially isomorphic; therefore, they are equivalent under the GMA or projection aberration criterion. However, they are not model-isomorphic and have different eligibilities. All three-factor and four-factor projections are eligible for the three designs, but the numbers of ineligible five-factor projections for $\mathbf{d}_4$, $\mathbf{d}_5$, and $\mathbf{d}_6$ are 3, 1, and 0, respectively. This example shows that a large number of model-nonisomorphic designs can be constructed through level permutations for some columns. The question of finding best level settings will be discussed in Section 3.2.

3.2 Computational Algorithms

While the previous combinatorial method has the advantage that the properties and structures of its constructed designs can be studied theoretically, there are other and sometimes better nonregular designs that cannot be obtained from using this method. Our second construction method is via computational algorithms.

To generate more OAs efficiently, we adopt an efficient algorithm due to Xu (2002), downloadable at `http://www.stat.ucla.edu/~hqxu`. This algorithm enables us to construct many 18- and 27-run OAs with different projection properties efficiently. For example, we found an $OA(27, 3^{13})$ (given in Table 11(ii)) that has 100% eligible three-, four- and five-factor projections. To our knowledge, none of the combinatorial methods described in Dey and Mukerjee (1999) and Hedayat, Sloane, and Stufken (1999) can produce such a design. Indeed, any 27-run design constructed by the combinatorial method described in the previous section must have some three-letter words if it has more than eight factors; therefore, some of the three-, four-, five-factor projections would be ineligible.

We also propose algorithms to determine the best level permutations. As mentioned earlier, we need to consider three permutations for a three-level factor. For n factors, there are 3^n possible level permutations. When n is not large, a complete search can be performed to determine the best level permutation. Otherwise, the following two column-wise greedy methods can be used.

Method I. (Sequential greedy) Start with the first column and permute three levels in that column with other columns fixed. Find and fix the best level setting for that column. Go to the next column or the first column if it is the last column. Repeat this procedure until no improvement is seen for n consecutive times.

Method II. (Random greedy) Randomly select one column and permute three levels in that column with other columns fixed. Find and fix the best level setting for that column. Repeat this procedure until no improvement is seen for some consecutive times, say $k = 10$.

Example 8. Consider the level permutation of $\mathbf{d}_4$ in Example 7. The results are shown in Table 4. The original design $\mathbf{d}_4$ has three (out of 56) ineligible five-factor projections (see $E_5 = 53$ in the row "no permutation"). Three methods find three different combinations of level permutations, which are listed under the column "level permutation". In a combination, each permutation is applied to a column in $\mathbf{d}_4$. For example, in Table 4, the best combination of level permutations in the complete search is (p_0, p_0, p_0, p_0, p_0, p_1, p_2, p_1). It means that no level permutation is required for columns

Table 4: A Comparison of Algorithms for Level Permutations

Method	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$	Seconds	Level Permutation
No Permutation	**56**	**70**	53	0.891	0.767	0.595	0	$p_0\ p_0\ p_0\ p_0\ p_0\ p_0\ p_0\ p_0$
Sequential Greedy	**56**	**70**	**56**	0.892	0.772	0.609	58	$p_2\ p_2\ p_2\ p_0\ p_0\ p_0\ p_0\ p_0$
Random Greedy	**56**	**70**	**56**	0.892	0.769	0.601	74	$p_0\ p_2\ p_0\ p_1\ p_0\ p_1\ p_0\ p_0$
Complete Search	**56**	**70**	**56**	0.892	0.772	0.609	12801	$p_0\ p_0\ p_0\ p_0\ p_0\ p_1\ p_2\ p_1$

1–5, the 6th and 8th columns should be permuted by p_1, and the 7th column by p_2. The design found by the complete search is the same as $\mathbf{d}_6$ constructed in Example 7. All methods succeed in finding level permutations such that all 56 five-factor projections are eligible (i.e., $E_5 = 56$). After level permutations, the estimation efficiencies also increase (e.g., $\bar{D}_3$ increases from 0.891 to 0.892). The complete search takes more than 200 minutes on a Sun Sparc workstation with 400M CPU while the sequential or random greedy search takes only about one minute.

4 Optimal Designs

In this section, we use the combinatorial and algorithmic methods to construct many new OAs with 18 and 27 runs, and then apply the three-step procedure to compare and rank them.

4.1 18-Run Designs

An 18-run OA can screen up to seven three-level factors. CW studied the commonly used OA given in Table 10(i) under the projection-efficiency criteria. A question is whether the design is optimal within a broader class of OAs.

First, by using Xu's (2002) algorithm, we generate 1,000 $OA(18, 3^7)$'s randomly. All these OAs have the same generalized wordlength pattern $(0, 0, 22, 34.5, 27, 31, 6)$; therefore, the GMA criterion cannot distinguish between them. Then we apply the projection aberration criterion and find that they fall into three classes. For illustration, we list three OAs in Table 10, each representing one class. The design in Table 10(i) is from standard textbooks (e.g., Wu and Hamada 2000, p. 335), and the other two constructed via the algorithm are chosen arbitrarily.

Table 5 shows the overall A_3's, projection frequencies, E_i's, and $\bar{D}_i$'s for the three arrays given in Table 10. All three arrays have one three-factor projection that has nine distinct runs (i.e.,

Table 5: Comparison of $OA(18, 3^7)$

	Overall	Proj. A_3 Freq.									
Array	A_3	$\frac{1}{2}$	$\frac{2}{3}$	1	2	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$
(i)	22	28	0	6	1	34	31	0	0.876	0.704	0
(ii)	22	20	12	2	1	34	28	0	0.871	0.684	0
(iii)	22	16	18	0	1	34	31	0	0.876	0.689	0

$A_3 = 2$). The projection aberration criterion would choose array (iii) because it has no projection with $A_3 = 1$ and the other two arrays have at least two projections with $A_3 = 1$ (which has 15 distinct runs). However, the difference is not substantial. Because all arrays have one three-letter word (i.e., $A_3 = 2$), they have the same number of eligible three-factor projections (i.e., $E_3 = 34$). Both arrays (i) and (iii) have 31 (out of 35) eligible four-factor projections while array (ii) has 28. As explained in CW, for 18-run designs, the degrees of freedom are not enough to entertain any second-order model with five factors; therefore, any five-factor projection is ineligible. Level permutations can increase the eligibility and estimation efficiency of the projections. A complete search is performed for each array. The best level permutations are (p_0, p_0, p_0, p_0, p_0, p_0, p_0), (p_2, p_0, p_2, p_1, p_0, p_0, p_0), and (p_2, p_0, p_2, p_2, p_1, p_0, p_2) for arrays (i), (ii), and (iii), respectively (see Table 6, $n = 7$). After level permutations, they have the same eligibility ($E_3 = 34, E_4 = 31, E_5 = 0$) and similar estimation efficiency. For array (i), there is no improvement (because no permutation is done); for array (ii), $\bar{D}_3$ increases from 0.871 to 0.881 and $\bar{D}_4$ increases from 0.684 to 0.694; for array (iii), $\bar{D}_4$ increases from 0.689 to 0.692. Therefore, after level permutations, these three arrays are competitive under the projection-efficiency criteria. It is interesting to point out that the three 4-factor projections $(2, 4, 5, 6)$, $(2, 4, 5, 7)$, and $(3, 5, 6, 7)$ from array (ii) are ineligible even if they are free of 3-letter words. They become eligible after level permutation (i.e., E_4 increases from 28 to 31).

Next, we consider subdesigns from these arrays. For each array, we search for the best n-factor subdesigns by applying the three-step procedure for all n with $3 \leq n \leq 7$. In the construction, Step 1 keeps all subdesigns with smallest overall A_3 values, Step 2 selects one best design under projection aberration criterion, and Step 3 uses a complete search. Table 6 shows the chosen designs, the projection properties, and level permutations for each n. For example, if a four-factor design is required from array (iii), we should choose columns 1, 3, 5, and 6 according to Table 6(iii).

Table 6: 18-Run Optimal Designs from Table 10

		Overall	Proj. A_3 Freq.											
Array	n	A_3	$\frac{1}{2}$	$\frac{2}{3}$	1	2	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$	Columns	Level Permutation
(i)	3	0.5	1	0	0	0	**1**	0	0	0.89	0	0	2 3 4	p_0 p_0 p_0
(i)	4	2	4	0	0	0	**4**	**1**	0	0.89	0.74	0	2 3 4 5	p_0 p_0 p_0 p_0
(i)	5	5	10	0	0	0	**10**	**5**	0	0.89	0.74	0	2 3 4 5 6	p_0 p_0 p_0 p_0 p_0
(i)	6	10	20	0	0	0	**20**	**15**	0	0.89	0.74	0	2 3 4 5 6 7	p_0 p_0 p_0 p_0 p_0 p_0
(i)	7	22	28	0	6	1	34	31	0	0.88	0.70	0	1 2 3 4 5 6 7	p_0 p_0 p_0 p_0 p_0 p_0 p_0
(ii)	3	0.5	1	0	0	0	**1**	0	0	0.89	0	0	1 2 5	p_0 p_0 p_1
(ii)	4	2	4	0	0	0	**4**	**1**	0	0.89	0.74	0	2 3 4 6	p_0 p_0 p_0 p_1
(ii)	5	5.67	6	4	0	0	**10**	**5**	0	0.89	0.71	0	2 3 4 5 6	p_0 p_0 p_1 p_0 p_2
(ii)	6	11.33	12	8	0	0	**20**	**15**	0	0.89	0.71	0	2 3 4 5 6 7	p_0 p_1 p_0 p_0 p_2 p_2
(ii)	7	22	20	12	2	1	34	31	0	0.88	0.69	0	1 2 3 4 5 6 7	p_2 p_0 p_2 p_1 p_0 p_0 p_0
(iii)	3	0.5	1	0	0	0	**1**	0	0	0.89	0	0	1 2 4	p_0 p_0 p_0
(iii)	4	2	4	0	0	0	**4**	**1**	0	0.89	0.74	0	1 3 5 6	p_0 p_1 p_0 p_2
(iii)	5	6	4	6	0	0	**10**	**5**	0	0.88	0.71	0	1 2 3 4 5	p_1 p_1 p_0 p_2 p_2
(iii)	6	12	8	12	0	0	**20**	**15**	0	0.88	0.71	0	1 2 3 4 5 6	p_1 p_0 p_0 p_1 p_1 p_2
(iii)	7	22	16	18	0	1	34	31	0	0.88	0.69	0	1 2 3 4 5 6 7	p_2 p_0 p_2 p_2 p_1 p_0 p_2

These four columns form an $OA(18, 3^4)$ which has minimum overall A_3 value of 2 and minimum projection aberration. Its levels should be permuted as (p_0, p_1, p_0, p_2). After the level permutations, its eligibility and efficiency are $E_3 = 4, E_4 = 1$ and $\bar{D}_3 = 0.89, \bar{D}_4 = 0.74$.

From Table 6, we observe that the best subdesigns (for $3 \le n \le 6$) can always be found from array (i) for all the criteria. Indeed, this is supported by a theoretical result. Applying Theorem 2 and Corollary 4 in Xu (2003), it can be shown that any subdesign not containing the first column of Table 10(i) has GMA and minimum projection aberration among *all* possible designs. In addition, all three- and four-factor projections not containing the first column are eligible and have the same estimation efficiencies.

In summary, the commonly used OA given in Table 10(i) and its subdesigns not containing the first column are recommended because any subdesign not containing the first column is optimal under the GMA and projection aberration criteria.

4.2 27-Run Designs

A 27-run OA can screen up to 13 three-level factors. CW studied the commonly used 27-run regular designs under the projection-efficiency criteria. They showed that minimum aberration designs are optimal *among* regular designs. They also gave an example to illustrate that nonregular designs may have higher projection-efficiency and are thus better than regular designs. Here we apply the construction methods in Section 3 and the three-step procedure in Section 2 to search for optimal designs in a much broader class of OAs.

First, we arbitrarily generate 100 $OA(27, 3^{13})$'s by applying the construction methods in Section 3. According to Xu and Wu (2001), all these (saturated) OAs have the same generalized wordlength pattern $(0, 0, 104, 468, \ldots)$; therefore, they are indistinguishable under the GMA criterion. Next we consider the projection aberration and projection-efficiency criteria. For illustration, we compare three designs: the regular 3^{13-10} design, the nonregular $OA(27, 3^{13})$ constructed in Example 6, and a nonregular $OA(27, 3^{13})$ constructed via Xu's algorithm. The two nonregular designs are given in Table 11 and referred to as nonregular design (i) and (ii), respectively. Table 7 shows the projection properties for the three arrays. The regular design has a simple projection pattern: A three-factor projection is either a 3^3 full factorial (i.e., projected $A_3 = 0$) or a 3^{3-1} design (i.e., projected $A_3 = 2$). Among the 286 three-factor projections, the regular design has 52 projections with three-letter words (i.e., projected $A_3 = 2$) and therefore has 52 ineligible three-

Table 7: Comparison of $OA(27, 3^{13})$

Array	Overall A_3	Projected A_3 Frequency 0	$\frac{8}{27}$	$\frac{4}{9}$	$\frac{14}{27}$	$\frac{2}{3}$	$\frac{20}{27}$	$\frac{10}{9}$	2	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$
Regular	104	234	0	0	0	0	0	0	52	234	234	0	0.93	0.86	0
Nonregular (i)	104	162	0	54	0	27	0	27	16	270	567	693	0.90	0.79	0.61
Nonregular (ii)	104	78	0	156	0	52	0	0	0	**286**	**715**	**1287**	0.90	0.78	0.62

factor projections (and $E_3 = 234$); the nonregular design (i) has 16 projections with three-letter words and therefore has 16 ineligible three-factor projections (and $E_3 = 270$); the nonregular design (ii) has no projection with three-letter word and therefore has no ineligible three-factor projection (and $E_3 = 286$). The number of eligible three-, four-, and five-factor projections are 234(82%), 234(33%), and 0(0%) for the regular design; 270(94%), 567(79%), and 693(54%) for the nonregular design (i); and 286(100%), 715(100%), and 1287(100%) for the nonregular design (ii), respectively. Therefore, under both the projection aberration and projection-efficiency criteria, the regular design is the worst and the nonregular design (ii) is the best. The nonregular design (ii) has the property that all its three-, four-, and five-factor projections are eligible and the estimation efficiencies are $\bar{D}_3 = 0.90$, $\bar{D}_4 = 0.78$, and $\bar{D}_5 = 0.62$. Now consider level permutations. For the regular design, the eligibility cannot be improved because of its three-letter words; for the nonregular design (i), the eligibility of five-factor projection is improved from $E_5 = 693$ to $E_5 = 714$ after the level permutations given in Table 8(i) under $n = 13$; for the nonregular design (ii), the eligibility cannot be improved (because it is already maximized) while the estimation efficiency $\bar{D}_5$ is slightly improved from 0.62 to 0.63 after the level permutations given in Table 8(ii) under $n = 13$.

Next, we consider subdesigns from the two nonregular OAs. For each array, we search for the best n-factor subdesigns by applying the three-step procedure for all n with $4 \leq n \leq 13$. In the construction, Step 1 keeps all subdesigns with smallest overall A_3 values, Step 2 selects one best design under projection aberration, and Step 3 uses a complete search if $n < 9$ and either a sequential or a random greedy search (with $k = 10$ tries) if $n \geq 9$. Table 8 shows the chosen designs, the projection properties, and level permutations for each n. For example, if an eight-factor design is required from Table 11(ii), we shall choose columns 1-4, 6-7, 11, and 13 according to Table 8(ii). The eight columns form an $OA(27, 3^8)$ which has the smallest overall A_3 value of 19.11 and minimum projection aberration (among all eight-factor subdesigns from Table 11(ii)).

Among its 56 three-factor projections, 17 projections are full factorials (i.e. $A_3 = 0$), 31 projections have an A_3 value of 4/9, and eight projections have an A_3 value of 2/3. From Table 8(ii), the level permutations are (p_0, p_1, p_0, p_1, p_2, p_0, p_0, p_1). The projection-efficiency of the resulting design is $E_3 = 56$, $E_4 = 70$, $E_5 = 56$, $\bar{D}_3 = 0.90$, $\bar{D}_4 = 0.79$, and $\bar{D}_5 = 0.64$. Note that this design has less projection aberration and higher projection-efficiency than all $OA(27, 3^8)$'s considered earlier in Examples 7 and 8.

It is interesting to compare the chosen designs in Table 8(i) and (ii). For $n = 4$, the two designs are equivalent (to a regular 3^{4-1} design). For $n = 11, 12, 13$, the designs from both tables have the same overall A_3 values, and the designs from Table 8(ii) have less projection aberration and better projection-efficiency than those from Table 8(i). Therefore, designs from Table 8(ii) are recommended. For $5 \leq n \leq 10$, the situation is more complicated. The designs from Table 8(ii) have larger overall A_3 values, less projection aberration and better projection-efficiency than those from Table 8(i). The choice of these designs depends on the objective. If factor screening is the primary task, we shall choose designs from Table 8(i) because they have smaller overall A_3 values. If interaction detection is the primary task, we shall choose designs from Table 8(ii) because all their three-, four-, and five-factor projections are eligible and have high efficiency.

In the preparation of the manuscript, we learned that there are exactly 68 combinatorially-nonisomorphic $OA(27, 3^{13})$'s (Lam and Tonchev 1996). Therefore, we further searched for optimal designs from all these 68 arrays. We found that the nonregular design (ii) given in Table 11(ii) has minimum projection aberration and that the designs given in Table 8(i) have GMA among all possible subdesigns from these 68 saturated OAs.

Because many 27-run OAs are not part of any saturated OA, we used algorithms to search for optimal designs. For each n, $4 \leq n \leq 12$, we constructed 1,000 $OA(27, 3^n)$'s by using Xu's algorithm and ranked them according to their overall A_3 values and projected A_3 values. We observed that their overall A_3 values are always larger than or equal to those given in Table 8(i). In other words, the designs given in Table 8(i) have minimum contamination. We also found many new OAs that are *not* part of any saturated OA and have less projection aberration than those given in Table 8 for $5 \leq n \leq 10$. Table 12 lists the best $OA(27, 3^n)$ under the projection aberration criterion for each n and Table 9 shows their projection properties. Level permutation algorithms have been applied to improve the projection-efficiency. Compared to the designs given in Table 8(ii), these new designs have slightly larger (i.e., worse) overall A_3 values, less (i.e., better) projection aberration, the same eligibility, and similar efficiency.

In summary, Table 11(ii) and its subdesigns given in Table 8(ii) are recommended because all their three-, four-, and five-factor projections are eligible and have high efficiencies.

5 Summary and Further Remarks

For factor screening and interaction detection, we propose computationally efficient criteria for ranking three-level designs. We show that the generalized wordlength pattern is closely related to various design properties: contamination, eligibility and estimation efficiency. In the three-step approach, these criteria are combined to sequentially search for optimal designs. Although we focus on three-level designs, this approach can be applied to designs with any number of levels because the generalized wordlength pattern is not restricted to three levels. In order to obtain more OAs for comparison, both combinatorial and algorithmic construction methods are proposed, and two versions of a search algorithm are presented for level permutations. Some 18- and 27-run optimal designs are found.

In the paper, only A_3 values (overall and projected) are used in ranking designs. For designs with small run size, use of the A_3 values suffices for discriminating and ranking designs. Because lower-dimensional projections are more important than higher-dimensional projections, there is no need to use A_4 values when A_3 can do the job. However, for large run size (e.g., 81), there are many designs with zero overall and projected A_3 values. These designs are equally good under the current contamination and projection aberration criteria. In this case, the projection aberration criterion should be modified and extended to A_4 values. The extension of projection frequency to A_4 values is straightforward. Because the overall A_4 value is not related to the aliasing between main effects and two-factor interactions, there is no need to extend the contamination criterion to A_4. Therefore, Step 1 in the three-step approach should be dropped.

For the construction of OAs, the algorithmic approach in Section 3.2 is very flexible and effective for small run size, such as 18 and 27, and outperforms the combinatorial method. However, it is computationally prohibitive (or even infeasible) for larger run size like 81. In this situation, combinatorial construction should be used until the algorithm can be further improved.

Acknowledgments

S.-W. Cheng's research was supported by the National Science Council of Taiwan, ROC; C. F. J. Wu's research was supported by NSF Grant DMS-0072489 and NSA Grant MSPR-00G-091; and H. Xu's research was supported by NSF Grant DMS-0204009. H. Xu is grateful to Professor Clement Lam for sending the list of 68 27-run orthogonal arrays. The authors thank the editor, the associate editor, and the referees for their criticism and valuable comments which contributed to the improvement of the paper.

Appendix: Proof of Lemma 2

Proof. Let $\mathbf{d}$ be a three-factor projection of an OA with N runs and s levels, denoted by $0, 1, \ldots, s-1$. For $i, j, k = 0, 1, \ldots, s-1$, let $f(i, j, k)$ be the number of times that the level combination (i, j, k) appears in $\mathbf{d}$. It is clear that $\sum_{k=0}^{s-1} f(i, j, k) = Ns^{-2}$ for $i, j = 0, \ldots, s-1$ since $\mathbf{d}$ is an OA. Then

$$\sum_{i=0}^{s-1}\sum_{j=0}^{s-1}\sum_{k=0}^{s-1} f(i,j,k)^2 \leq \sum_{i=0}^{s-1}\sum_{j=0}^{s-1}\left[\sum_{k=0}^{s-1} f(i,j,k)\right]^2 = \sum_{i=0}^{s-1}\sum_{j=0}^{s-1}\left[Ns^{-2}\right]^2 = N^2 s^{-2},$$

where the equality holds if and only if for each pair of i, j, there is a unique k such that $f(i, j, k) = Ns^{-2}$. In other words, the equality holds if and only if the factor levels of the first two columns completely determine the factor level of the third column, i.e., the three factors form a three-letter word. On the other hand, let $B_0 = N^{-1}\sum_{i=0}^{s-1}\sum_{j=0}^{s-1}\sum_{k=0}^{s-1} f(i, j, k)^2$. By applying some fundamental results from coding theory, Xu and Wu (2001) showed that B_0 is a linear combination of the generalized wordlength pattern. Specifically, for a design of N runs and three s-level factors, $B_0 = Ns^{-3}(1 + A_1 + A_2 + A_3)$. From Xu and Wu (2001), $A_1 = A_2 = 0$ for an OA. Therefore, $A_3 = N^{-1}s^3 B_0 - 1 \leq s - 1$, where the equality holds if and only if the three factors form a three-letter word. □

References

Box, G. E. P., and Draper, N. R. (1987), *Empirical Model-Building and Response Surfaces*, New York: Wiley.

Box, G. E. P., and Meyer, R. D. (1986). "An Analysis for Unreplicated Fractional Factorials," *Technometrics*, 28, 11–18.

—— (1993), "Finding the Active Factors in Fractionated Screening Experiments," *Journal of Quality Technology*, 25, 94–105.

Cheng, S. W., and Wu, C. F. J. (2001), "Factor Screening and Response Surface Exploration (With Discussion)," *Statistica Sinica*, 11, 553–604.

Chipman, H. (1996), "Bayesian Variable Selection With Related Predictors," *The Canadian Journal of Statistics*, 24, 17–36.

Chipman, H., Hamada, M., and Wu, C.F.J. (1997), "A Bayesian Variable-Selection Approach for Analyzing Designed Experiments With Complex Aliasing," *Technometrics*, 39, 372–381.

Deng, L. Y., and Tang, B. (2002), "Design Selection and Classification for Hadamard Matrices Using Generalized Minimum Aberration Criteria," *Technometrics*, 44, 173–184.

Dey, A., and Mukerjee, R. (1999), *Fractional Factorial Plans*, New York: Wiley.

Farrell, R. H., Kiefer, J., and Walbran, A. (1967), "Optimum Multivariate Designs," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. I: Statistics*, eds. L. M. L. Cam, and J. Neyman, California: University of California Press, pp. 113–138.

Fries, A., and Hunter, W. G. (1980), "Minimum Aberration 2^{k-p} Designs," *Technometrics*, 22, 601–608.

Hamada, M., and Wu, C. F. J. (1992), "Analysis of Designed Experiments With Complex Aliasing," *Journal of Quality Technology*, 24, 130–137.

Hedayat, A. S., Sloane, N. J. A., and Stufken, J. (1999), *Orthogonal Arrays: Theory and Applications*, New York: Springer-Verlag.

Khuri, A., and Cornell, J. A. (1996), *Response Surfaces: Designs and Analyses*, 2nd edition, New York: Marcel Dekker.

Kiefer, J. (1961), "Optimum Designs in Regression Problems II," *The Annals of Mathematical Statistics*, 32, 298–325.

King, C., and Allen, L. (1987), "Optimization of Winding Operation for Radio Frequency Chokes," in *Fifth Symposium on Taguchi Methods,* Dearborn, Michigan: American Supplier Institute, pp. 67–80.

Lam, C., and Tonchev, V. D. (1996), "Classification of Affine Resolvable 2-(27,9,4) Designs," *Journal of Statistical Planning and Inference*, 56, 187–202.

McCullagh, P., and Nelder, J. A. (1989), *Generalized Linear Models*, 2nd ed, London: Chapman and Hall.

Myers, R. H., and Montgomery, D. C. (1995), *Response Surface Methodology: Process and Product in Optimization Using Designed Experiments*, New York: Wiley.

Steinberg, D. M., and Bursztyn D. (2001), Discussion of "Factor Screening and Response Surface Exploration," by S.-W. Cheng and C. F. J. Wu, *Statistica Sinica*, 11, 596–599.

Tang, B., and Deng, L. Y. (1999), "Minimum G_2-Aberration for Non-Regular Fractional Factorial Designs," *The Annals of Statistics*, 27, 1914–1926.

Tsai, P. W., Gilmour, S. G., and Mead, R. (2000), "Projective Three-Level Main Effects Designs Robust to Model Uncertainty," *Biometrika*, 87, 467–475.

Wang, J. C., and Wu, C. F. J. (1991), "An Approach to the Construction of Asymmetrical Orthogonal Arrays," *Journal of the American Statistical Association*, 86, 450–456.

Wu, C. F. J., and Hamada, M. (2000), *Experiments: Planning, Analysis and Parameter Design Optimization*, New York: Wiley.

Xu, H. (2002), "An Algorithm for Constructing Orthogonal and Nearly Orthogonal Arrays With Mixed Levels and Small Runs," *Technometrics*, 44, 356–368.

—— (2003), "Minimum Moment Aberration for Nonregular Designs and Supersaturated Designs," *Statistica Sinica*, to appear.

Xu, H., and Wu, C. F. J. (2001), "Generalized Minimum Aberration for Asymmetrical Fractional Factorial Designs," *The Annals of Statistics*, 29, 1066–1077.

Table 8: 27-Run Optimal Designs from Table 11

Array	n	Overall A_3	Projected A_3 Frequency 0	$\frac{8}{27}$	$\frac{4}{9}$	$\frac{14}{27}$	$\frac{2}{3}$	$\frac{20}{27}$	$\frac{10}{9}$	2	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$	Columns	Level Permutation
(i)	4	0	4	0	0	0	0	0	0	0	**4**	**1**	0	0.93	0.88	0	1 2 5 6	$p_0\ p_0\ p_0\ p_1$
(i)	5	2	6	0	3	0	1	0	0	0	**10**	**5**	**1**	0.91	0.83	0.70	1 2 4 7 13	$p_0\ p_0\ p_0\ p_1\ p_1$
(i)	6	4	18	0	0	0	0	0	0	2	18	9	0	0.93	0.88	0	1 2 5 6 9 10	$p_0\ p_0\ p_0\ p_1\ p_0\ p_2$
(i)	7	10	22	0	0	0	12	0	0	1	34	31	15	0.91	0.81	0.65	1 2 4 5 6 8 9	$p_0\ p_0\ p_1\ p_1\ p_0\ p_2\ p_1$
(i)	8	16	36	0	0	0	18	0	0	2	54	60	36	0.91	0.81	0.65	1 2 4 5 6 8 9 10	$p_0\ p_0\ p_2\ p_1\ p_0\ p_1\ p_2\ p_0$
(i)	9	24	54	0	0	0	27	0	0	3	81	108	81	0.91	0.81	0.65	1 2 4 5 6 8 9 10 12	$p_0\ p_0\ p_1\ p_2\ p_0\ p_2\ p_1\ p_0\ p_0$
(i)	10	42	63	0	18	0	27	0	9	3	117	189	180	0.90	0.79	0.62	1 2 3 4 5 6 8 9 10 12	$p_0\ p_1\ p_1\ p_0\ p_0\ p_0\ p_0\ p_0\ p_0\ p_1$
(i)	11	60	81	0	36	0	27	0	18	3	162	306	367	0.90	0.78	0.61	1 2 3 4 5 6 7 8 9 10 12	$p_0\ p_0\ p_1\ p_0\ p_0\ p_1\ p_1\ p_0\ p_0\ p_0\ p_1$
(i)	12	80	108	0	54	0	27	0	27	4	216	459	633	0.90	0.78	0.61	1 2 3 4 5 6 7 8 9 10 11 12	$p_0\ p_0\ p_0\ p_1\ p_0\ p_0\ p_0\ p_1\ p_0\ p_1\ p_0\ p_0$
(i)	13	104	162	0	54	0	27	0	27	16	270	567	714	0.90	0.79	0.61	1 2 3 4 5 6 7 8 9 10 11 12 13	$p_0\ p_0\ p_0\ p_0\ p_2\ p_1\ p_0\ p_0\ p_2\ p_0\ p_0\ p_0\ p_1$
(ii)	4	0	4	0	0	0	0	0	0	0	**4**	**1**	0	0.93	0.88	0	1 2 7 8	$p_0\ p_0\ p_0\ p_0$
(ii)	5	2.44	5	0	4	0	1	0	0	0	**10**	**5**	**1**	0.92	0.83	0.71	1 2 3 7 8	$p_0\ p_1\ p_2\ p_2\ p_1$
(ii)	6	6	8	0	9	0	3	0	0	0	**20**	**15**	**6**	0.91	0.81	0.65	1 2 3 4 8 9	$p_0\ p_2\ p_2\ p_2\ p_0\ p_2$
(ii)	7	11.33	12	0	18	0	5	0	0	0	**35**	**35**	**21**	0.90	0.8	0.65	1 2 3 4 8 9 11	$p_2\ p_0\ p_0\ p_1\ p_0\ p_0\ p_0$
(ii)	8	19.11	17	0	31	0	8	0	0	0	**56**	**70**	**56**	0.90	0.79	0.64	1 2 3 4 6 7 11 13	$p_0\ p_1\ p_0\ p_1\ p_2\ p_0\ p_0\ p_1$
(ii)	9	29.33	24	0	48	0	12	0	0	0	**84**	**126**	**126**	0.90	0.79	0.64	1 2 3 4 5 6 7 11 13	$p_1\ p_0\ p_0\ p_0\ p_0\ p_1\ p_1\ p_0\ p_1$
(ii)	10	43.33	33	0	66	0	21	0	0	0	**120**	**210**	**252**	0.90	0.79	0.63	1 2 3 4 5 6 7 8 10 12	$p_1\ p_0\ p_1\ p_0\ p_1\ p_0\ p_0\ p_0\ p_2\ p_0$
(ii)	11	60	45	0	90	0	30	0	0	0	**165**	**330**	**462**	0.90	0.78	0.63	1 2 3 4 5 6 7 8 9 10 11	$p_2\ p_0\ p_0\ p_0\ p_2\ p_0\ p_0\ p_0\ p_0\ p_0\ p_0$
(ii)	12	80	60	0	120	0	40	0	0	0	**220**	**495**	**792**	0.90	0.78	0.63	1 2 3 4 5 6 7 8 9 10 11 12	$p_0\ p_2\ p_0\ p_0\ p_2\ p_2\ p_0\ p_0\ p_1\ p_0\ p_1\ p_0$
(ii)	13	104	78	0	156	0	52	0	0	0	**286**	**715**	**1287**	0.90	0.78	0.63	1 2 3 4 5 6 7 8 9 10 11 12 13	$p_0\ p_0\ p_0\ p_0\ p_1\ p_1\ p_0\ p_1\ p_1\ p_0\ p_0\ p_0\ p_0$

Table 9: 27-Run Optimal Designs from Table 12

	Overall	Projected A_3 Frequency														
n	A_3	0	$\frac{8}{27}$	$\frac{4}{9}$	$\frac{14}{27}$	$\frac{2}{3}$	$\frac{20}{27}$	$\frac{10}{9}$	2	E_3	E_4	E_5	$\bar{D}_3$	$\bar{D}_4$	$\bar{D}_5$	Array
5	3.11	0	9	1	0	0	0	0	0	**10**	**5**	**1**	0.92	0.82	0.69	oa27.5
6	7.56	0	9	11	0	0	0	0	0	**20**	**15**	**6**	0.91	0.81	0.68	oa27.6
7	12.3	1	21	9	4	0	0	0	0	**35**	**35**	**21**	0.90	0.80	0.65	oa27.7
8	20.37	5	24	10	17	0	0	0	0	**56**	**70**	**56**	0.90	0.79	0.64	oa27.8
9	30.3	10	32	23	14	5	0	0	0	**84**	**126**	**126**	0.90	0.79	0.64	oa27.9
10	43.93	19	31	39	22	9	0	0	0	**120**	**210**	**252**	0.90	0.79	0.63	oa27.10

Table 10: $OA(18, 3^7)$

(i)								(ii)								(iii)							
Run	1	2	3	4	5	6	7	Run	1	2	3	4	5	6	7	Run	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0	1	0	2	0	0	1	2	2	1	0	1	0	2	1	0	2
2	0	1	1	1	1	1	1	2	1	1	0	1	0	2	1	2	1	0	1	2	0	0	1
3	0	2	2	2	2	2	2	3	2	2	0	1	2	0	0	3	2	2	2	0	1	0	1
4	1	0	0	1	1	2	2	4	0	0	2	0	0	0	0	4	0	0	2	1	1	1	0
5	1	1	1	2	2	0	0	5	1	0	1	1	1	1	0	5	1	1	2	0	0	2	0
6	1	2	2	0	0	1	1	6	2	0	1	2	0	2	2	6	2	0	1	0	2	1	2
7	2	0	1	0	2	1	2	7	0	2	1	2	2	1	1	7	0	2	1	1	0	2	2
8	2	1	2	1	0	2	0	8	1	1	2	2	2	0	2	8	1	1	0	1	2	1	1
9	2	2	0	2	1	0	1	9	2	1	2	0	1	1	1	9	2	2	0	2	2	2	0
10	0	0	2	2	1	1	0	10	0	1	0	2	0	1	0	10	0	0	2	2	2	2	1
11	0	1	0	0	2	2	1	11	1	2	1	0	0	0	1	11	1	0	0	0	1	2	2
12	0	2	1	1	0	0	2	12	2	1	1	0	2	2	0	12	2	1	1	1	1	2	1
13	1	0	1	2	0	2	1	13	0	1	1	1	1	0	2	13	0	1	1	0	2	0	0
14	1	1	2	0	1	0	2	14	1	2	2	2	1	2	0	14	1	2	2	1	2	0	2
15	1	2	0	1	2	1	0	15	2	2	2	1	0	1	2	15	2	1	2	2	0	1	2
16	2	0	2	1	2	0	1	16	0	0	2	1	2	2	1	16	0	2	0	0	0	1	1
17	2	1	0	2	0	1	2	17	1	0	0	0	2	1	2	17	1	2	1	2	1	1	0
18	2	2	1	0	1	2	0	18	2	0	0	2	1	0	1	18	2	0	0	1	0	0	0

Table 11: $OA(27, 3^{13})$

(i)

Run	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	1	2	0	1	1	2	0	1	1	2	0
3	0	2	2	1	0	2	2	1	0	2	2	1	0
4	1	0	1	1	1	0	1	1	1	0	1	1	0
5	1	1	2	0	1	1	2	0	1	1	2	0	0
6	1	2	0	2	1	2	0	2	1	2	0	2	0
7	2	0	2	2	2	0	2	2	2	0	2	2	0
8	2	1	0	1	2	1	0	1	2	1	0	1	0
9	2	2	1	0	2	2	1	0	2	2	1	0	0
10	0	0	2	1	1	1	0	2	2	2	1	0	1
11	0	1	1	0	1	2	2	1	2	0	0	2	1
12	0	2	0	2	1	0	1	0	2	1	2	1	1
13	1	0	1	2	2	1	2	0	0	2	0	1	1
14	1	1	0	1	2	2	1	2	0	0	2	0	1
15	1	2	2	0	2	0	0	1	0	1	1	2	1
16	2	0	0	0	0	1	1	1	1	2	2	2	1
17	2	1	2	2	0	2	0	0	1	0	1	1	1
18	2	2	1	1	0	0	2	2	1	1	0	0	1
19	0	0	0	0	2	2	2	2	1	1	1	1	2
20	0	1	1	2	2	0	0	1	1	2	2	0	2
21	0	2	2	1	2	1	1	0	1	0	0	2	2
22	1	0	1	1	0	2	0	0	2	1	2	2	2
23	1	1	2	0	0	0	1	2	2	2	0	1	2
24	1	2	0	2	0	1	2	1	2	0	1	0	2
25	2	0	2	2	1	2	1	1	0	1	0	0	2
26	2	1	0	1	1	0	2	0	0	2	1	2	2
27	2	2	1	0	1	1	0	2	0	0	2	1	2

NOTE: Constructed via combinatorial method in Section 3.1.

(ii)

Run	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	0	1	2	0	0	0	0	0	0
2	0	1	1	2	2	2	2	2	2	2	1	0	0
3	0	2	2	2	1	2	1	0	0	0	1	2	1
4	0	0	1	1	2	1	0	1	1	0	1	1	1
5	0	1	2	2	0	1	1	1	1	1	2	0	2
6	0	2	0	0	2	2	0	2	1	1	0	2	2
7	0	0	0	1	0	0	1	2	2	2	2	2	1
8	0	1	2	1	1	0	0	0	2	1	0	1	0
9	0	2	1	0	1	0	2	1	0	2	2	1	2
10	1	0	1	2	1	1	0	2	0	1	2	2	0
11	1	1	0	0	1	2	0	1	2	0	2	0	1
12	1	2	1	2	0	0	0	0	1	2	0	0	1
13	1	0	2	2	2	0	2	1	2	0	0	2	2
14	1	1	0	1	1	1	2	0	1	2	1	2	2
15	1	2	0	1	2	0	1	1	0	1	1	0	0
16	1	0	1	0	0	2	1	0	2	1	1	1	2
17	1	1	2	0	2	1	1	2	0	2	0	1	1
18	1	2	2	1	0	2	2	2	1	0	2	1	0
19	2	0	0	2	1	2	1	1	1	2	0	1	0
20	2	1	1	0	2	0	1	0	1	0	2	2	0
21	2	2	2	0	0	1	0	1	2	2	1	2	0
22	2	0	2	1	2	2	0	0	0	2	2	0	2
23	2	1	1	1	0	2	2	1	0	1	0	2	1
24	2	2	0	2	2	1	2	0	2	1	2	1	1
25	2	0	2	0	1	0	2	2	1	1	1	0	1
26	2	1	0	2	0	0	0	2	0	0	1	1	2
27	2	2	1	1	1	1	1	2	2	0	0	0	2

NOTE: Constructed via algorithmic method in Section 3.2.

Table 12: $OA(27, 3^n)$

oa27.5

Run	1	2	3	4	5
1	0	0	2	1	1
2	0	1	0	1	2
3	0	2	0	2	0
4	0	0	2	2	2
5	0	1	1	2	1
6	0	2	2	1	0
7	0	0	0	0	1
8	0	1	1	0	0
9	0	2	1	0	2
10	1	0	0	1	0
11	1	1	2	0	2
12	1	2	2	2	1
13	1	0	0	2	2
14	1	1	2	0	1
15	1	2	0	0	1
16	1	0	1	2	0
17	1	1	1	1	0
18	1	2	1	1	2
19	2	0	1	0	2
20	2	1	0	2	2
21	2	2	1	2	1
22	2	0	1	1	1
23	2	1	2	2	0
24	2	2	0	0	0
25	2	0	2	0	0
26	2	1	0	1	1
27	2	2	2	1	2

oa27.6

Run	1	2	3	4	5	6
1	0	0	1	0	0	1
2	0	1	1	2	2	1
3	0	2	2	2	0	1
4	0	0	0	0	2	2
5	0	1	0	1	0	0
6	0	2	1	0	1	0
7	0	0	2	2	1	2
8	0	1	2	1	2	0
9	0	2	0	1	1	2
10	1	0	0	2	0	0
11	1	1	2	0	1	1
12	1	2	2	0	2	2
13	1	0	2	1	1	0
14	1	1	1	1	0	2
15	1	2	0	2	2	0
16	1	0	1	1	1	1
17	1	1	1	2	2	2
18	1	2	0	0	0	1
19	2	0	0	1	2	1
20	2	1	2	0	0	0
21	2	2	1	1	0	2
22	2	0	1	0	2	0
23	2	1	0	0	1	2
24	2	2	2	1	2	1
25	2	0	2	2	0	2
26	2	1	0	2	1	1
27	2	2	1	2	1	0

oa27.7

Run	1	2	3	4	5	6	7
1	0	0	1	2	2	0	1
2	0	1	0	1	2	1	0
3	0	2	2	0	2	1	2
4	0	0	0	0	1	2	0
5	0	1	1	2	1	2	2
6	0	2	1	0	0	0	0
7	0	0	2	1	0	2	1
8	0	1	2	2	0	0	2
9	0	2	0	1	1	1	1
10	1	0	0	2	2	1	2
11	1	1	2	0	1	1	1
12	1	2	2	2	2	2	1
13	1	0	0	0	0	0	1
14	1	1	0	2	0	2	0
15	1	2	1	1	1	0	2
16	1	0	1	1	0	1	2
17	1	1	2	1	1	0	0
18	1	2	1	0	2	2	0
19	2	0	1	2	1	1	0
20	2	1	1	0	0	1	1
21	2	2	0	2	1	0	1
22	2	0	2	0	1	2	2
23	2	1	0	0	2	0	2
24	2	2	0	1	0	2	2
25	2	0	2	1	2	0	0
26	2	1	1	1	2	2	1
27	2	2	2	2	0	1	0

oa27.8

Run	1	2	3	4	5	6	7	8
1	0	0	0	2	0	2	0	1
2	0	1	0	0	1	2	2	2
3	0	2	2	1	0	0	2	0
4	0	0	1	0	0	1	1	1
5	0	1	1	1	2	0	0	1
6	0	2	2	0	2	2	0	0
7	0	0	2	2	2	0	1	2
8	0	1	1	2	1	1	2	0
9	0	2	0	1	1	1	1	2
10	1	0	2	1	1	1	0	2
11	1	1	2	2	0	2	2	2
12	1	2	1	0	1	0	2	1
13	1	0	0	0	1	0	0	0
14	1	1	1	1	2	2	1	0
15	1	2	1	2	2	1	0	2
16	1	0	0	1	2	2	2	1
17	1	1	2	0	0	1	1	1
18	1	2	0	2	0	0	1	0
19	2	0	1	2	1	2	1	0
20	2	1	0	1	0	1	0	0
21	2	2	0	2	2	1	2	1
22	2	0	2	0	2	1	2	0
23	2	1	2	2	1	0	0	1
24	2	2	1	0	0	2	0	2
25	2	0	1	1	0	0	2	2
26	2	1	0	0	2	0	1	2
27	2	2	2	1	1	2	1	1

oa27.9

Run	1	2	3	4	5	6	7	8	9
1	0	0	2	2	1	2	2	1	1
2	0	1	2	1	0	1	1	0	2
3	0	2	0	2	0	2	0	2	2
4	0	0	1	1	2	0	0	2	2
5	0	1	1	0	2	1	2	0	1
6	0	2	1	0	0	0	1	1	1
7	0	0	0	0	1	2	1	0	0
8	0	1	0	1	1	0	2	2	0
9	0	2	2	2	2	1	0	1	0
10	1	0	1	2	2	1	1	2	0
11	1	1	0	2	1	1	1	1	2
12	1	2	0	2	2	0	2	0	1
13	1	0	2	1	0	0	2	1	0
14	1	1	2	0	2	2	2	2	2
15	1	2	2	0	1	0	0	0	2
16	1	0	0	1	0	1	0	0	1
17	1	1	1	0	0	2	0	1	0
18	1	2	1	1	1	2	1	2	1
19	2	0	0	0	2	0	1	1	2
20	2	1	2	2	0	0	1	2	1
21	2	2	2	1	2	2	1	0	0
22	2	0	1	2	0	2	2	0	2
23	2	1	0	1	2	2	0	1	1
24	2	2	1	1	1	1	2	1	2
25	2	0	2	0	1	1	0	2	1
26	2	1	1	2	1	0	0	0	0
27	2	2	0	0	0	1	2	2	0

oa27.10

Run	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	1	0	0	2	2	1
2	0	1	2	0	1	0	1	0	0	0
3	0	2	0	1	2	2	1	2	0	0
4	0	0	1	1	0	2	0	0	0	1
5	0	1	2	1	2	0	2	1	2	2
6	0	2	0	0	0	1	1	1	1	2
7	0	0	2	2	0	2	2	1	1	0
8	0	1	1	2	1	1	2	2	1	1
9	0	2	1	2	2	1	0	0	2	2
10	1	0	2	2	1	2	1	0	2	2
11	1	1	0	2	1	1	0	1	0	0
12	1	2	0	1	1	2	2	1	2	1
13	1	0	1	0	2	0	1	1	1	1
14	1	1	0	1	0	0	0	0	1	2
15	1	2	1	2	0	0	1	2	2	0
16	1	0	2	1	2	1	0	2	1	0
17	1	1	1	0	0	2	2	2	0	2
18	1	2	2	0	2	1	2	0	0	1
19	2	0	1	1	1	1	1	1	0	2
20	2	1	1	0	2	2	0	1	2	0
21	2	2	1	1	1	0	2	0	1	0
22	2	0	0	2	2	0	2	2	0	2
23	2	1	0	2	2	2	1	0	1	1
24	2	2	2	0	1	2	0	2	1	2
25	2	0	0	0	0	1	2	0	2	0
26	2	1	2	1	0	1	1	2	2	1
27	2	2	2	2	0	0	0	1	0	1

NOTE: Constructed via algorithmic method in Section 3.2.