Algorithmic Construction of Efficient Fractional Factorial Designs With Large Run Sizes

Hongquan Xu

Department of Statistics

University of California

Los Angeles, CA 90095-1554

(hqxu@stat.ucla.edu)

September 28, 2007

Fractional factorial designs are widely used in practice and typically chosen according to the minimum aberration criterion. A sequential algorithm is developed for constructing efficient fractional factorial designs. A construction procedure is proposed that only allows a design to be constructed from its minimum aberration projection in the sequential build-up process. To efficiently identify nonisomorphic designs, designs are divided into different categories according to their moment projection patterns. A fast isomorphism check procedure is developed by matching the factors using their delete-one-factor projections. A method is proposed for constructing minimum aberration designs using only a partial catalog of some good designs. Minimum aberration designs are constructed for 128 runs up to 64 factors, 256 runs up to 28 factors, and 512, 1024, 2048, and 4096 runs up to 23 or 24 factors. Furthermore, this algorithm is used to completely enumerate all 128-run designs of resolution 4 up to 30 factors, all 256-run designs of resolution 4 up to 17 factors, all 512-run designs of resolution 5, all 1024-run designs of resolution 6, and all 2048- and 4096-run designs of resolution 7.

KEY WORDS: Fractional factorial design, isomorphism, linear code, MacWilliams identity, minimum aberration, resolution

1 Introduction

Fractional factorial (FF) designs are widely used in many areas of science, engineering and industry. With the rapidly increasing computational power, more and more large FF designs are used in large scale computer experiments where physical processes are being simulated. Researchers at Johns Hopkins University employed two-level FF designs in a ballistic missile defense project to assess the sensitivity of 47 parameters of an extended air defense simulation in two far-term scenarios over the first 10 days of a war (Mee 2004). In the first scenario, a resolution IV design with 512 runs was initially used and followed by 352 additional runs to resolve aliasing of two-factor interactions. The second scenario was explored using a resolution V design with 4,096 runs obtained by SAS's PROC FACTEX. A more economic design in the second scenario would have been a resolution V design with 2,048 runs, which can be obtained from a binary linear code constructed by Chen (1991). Lin and Sitter (2006) reported that FF designs with over 600 runs and as many as 53 parameters were used in computer simulations at Los Alamos National Laboratory.

FF designs are often chosen by the minimum aberration (MA) criterion (Fries and Hunter 1980), an extension of the maximum resolution criterion (Box and Hunter 1961); see, among others, Box, Hunter, and Hunter (2005, Table 6.22), Dean and Voss (1999, Tables 15.55 and 15.56), Montgomery (2005, Table 8-14), and Wu and Hamada (2000, Tables 4A and 5A). The reader is referred to Wu and Hamada (2000) and Mukerjee and Wu (2006) for rich results on MA designs and extensive references.

The construction of MA designs or optimal designs under other criteria is not straightforward, especially when the run size is large. Draper and Mitchell (1967, 1968) first developed a stage-by-stage algorithm and completely enumerated all 256-run designs of resolution ≥ 5 and all even 512-run designs of resolution ≥ 6. An even design contains entirely defining words of even length whereas an odd design has at least one defining word of odd length. Draper and Mitchell (1970) attempted but failed to construct the complete set of even 1024-run designs of resolution ≥ 6, and the complete set of odd 512-run designs of resolution ≥ 5. They obtained 4,043 distinct even 1024-run designs of resolution ≥ 6; as we will see later, they missed about 30% designs.

The construction of efficient FF designs is relatively easier when the run size is smaller. Chen, Sun and Wu (1993, CSW hereafter) developed a sequential algorithm and enumerated all 8, 16, 27, 32-run designs of resolution ≥ 3 and 64-run designs of resolution ≥ 4. Xu (2005) extended their work and enumerated all 81-run designs of resolution ≥ 3, 243-run designs of resolution ≥ 4, and 729-run designs of resolution ≥ 5. Based on a conjecture, Block and Mee (2005) constructed MA 128-run designs for 12 to 64 factors. Lin and Sitter (2006) developed an algorithm and enumerated all 128-run designs of resolution ≥ 4 up to 16 factors and all 512-run designs of resolution ≥ 5 up to 16 factors.

A key step in any algorithmic construction of FF designs is to determine whether two designs are isomorphic. Two FF designs are *isomorphic* (or *equivalent*) if and only if one may be obtained from the other by a relabeling of the factors. Two designs are distinct if they are not equivalent. For large FF designs, the test of equivalence of two designs requires an excessive amount of computer time, so many test procedures have been proposed to quickly identify nonisomorphic designs. Draper and Mitchell (1967) used the wordlength pattern to distinguish designs. Unfortunately, two nonisomorphic designs can have the same wordlength pattern, so Draper and Mitchell (1970) used a "letter pattern comparison" to test the equivalency of two designs and conjectured that FF designs with the same letter pattern are isomorphic. However, Chen and Lin (1991) disproved their conjecture by constructing two nonisomorphic 2^{31-15} designs with the same letter pattern. Zhu and Zeng (2005) reported that counter examples exist for as small as 32 runs; they also proposed a more sensitive test based on the coset pattern, which still fails to determine a design uniquely. Block and Mee (2005) conjectured that two designs are isomorphic if their sets of delete-one-factor projections are equivalent. See Clark and Dean (2001), Ma, Fang, and Lin (2001), Xu (2005), and Lin and Sitter (2006) for other test procedures.

In this paper we develop a new algorithm for constructing efficient FF designs with large run sizes. As in other algorithms, we construct designs sequentially by adding one factor at a time. We introduce an intelligent construction procedure that only allows a design to be constructed from its MA projection in the sequential build-up process. This procedure discards many isomorphic designs without performing time-consuming isomorphism checks. As we will see later, this procedure is more efficient than the procedure used by Lin and Sitter (2006) who adopted a combined approach from Bingham and Sitter (1999). To identify nonisomorphic designs, we divide designs into different categories according to their moment projection patterns. As demonstrated by Xu (2005), the use of moment projection patterns is more efficient than the use of letter patterns in terms of both distinguishing designs and computation. To test whether two designs in the same category are isomorphic, we develop a fast isomorphism check procedure by matching the factors using their delete-one-factor projections. This procedure skips many unsuccessful relabeling maps and is much more efficient than the procedures used by CSW and Lin and Sitter (2006). Based on an upper bound on the wordlength pattern, we propose a method for constructing MA designs using only a partial catalog of some good designs. The new algorithm enables us to construct MA designs for 128 runs up to 64 factors, 256 runs up to 28 factors, and 512, 1024, 2048 and 4096 runs up to 23 or 24 factors. Furthermore, we completely enumerate all 128-run designs of resolution ≥ 4 up to 30 factors, all 256-run designs of resolution ≥ 4 up to 17 factors, all 512-run designs of resolution ≥ 5, all 1024-run designs of resolution ≥ 6, and all 2048- and 4096-run designs of resolution ≥ 7. For clarity, we consider only two-level *regular* FF designs. The extension to multi-level designs is straightforward.

In Section 2, we review some basic concepts, definitions and preliminary results. We describe the construction method in Section 3. Tables of designs with 128–4096 runs are given in Section 4 and concluding remarks are given in Section 5.

2 Basic concepts, definitions and preliminary results

A regular 2^{n-k} FF design, denoted by D, has n factors of two levels and 2^{n-k} runs. A factor is also called a letter or a column whereas a run is called a row. Associated with every regular 2^{n-k} design is a set of k independent defining words. The defining contrast subgroup of D consists of all possible products of the k defining words and has 2^k words (including the identity I). Let $A_i(D)$ be the number of words of length i. The vector $(A_1(D), \ldots, A_n(D))$ is called the *wordlength pattern*. The *resolution* is the smallest i such that $A_i(D) > 0$.

Let D_1 and D_2 be two regular 2^{n-k} designs. D_1 is said to have less aberration than D_2 if there exists an r such that $A_i(D_1) = A_i(D_2)$ for $i = 1, \ldots, r-1$ and $A_r(D_1) < A_r(D_2)$. D_1 is said to have *minimum aberration* (MA) if there is no other regular design with less aberration than D_1.

A 2^{n-k} design D of resolution R is said to have *weak MA* (Chen and Hedayat 1996) if it has maximum resolution and $A_R(D)$ is minimized among all regular designs.

2.1 Connection with coding theory

The connection between factorial designs and linear codes is important in the development of our algorithm. For an introduction to coding theory, see Hedayat, Sloane, and Stufken (1999, Chapter 4) and MacWilliams and Sloane (1977).

A regular 2^{n-k} FF design D is also known as a linear code of length n and dimension $n-k$ over the binary field $GF(2)$ in coding theory. Associated with every binary linear code is another linear code, the dual code $D^\perp$, that consists of all row vectors $(u_1, \ldots, u_n)$ over $GF(2)$ such that $\sum_{i=1}^n u_i v_i = 0$ for all $(v_1, \ldots, v_n)$ in D.

The *Hamming weight* of a vector $(u_1, \ldots, u_n)$ is the number of nonzero components u_i. Let $B_i(D)$ and $B_i(D^\perp)$ be the number of rows with Hamming weight i in D and $D^\perp$, respectively. The vectors $(B_0(D), B_1(D), \ldots, B_n(D))$ and $(B_0(D^\perp), B_1(D^\perp), \ldots, B_n(D^\perp))$ are called the *weight distributions* of D and $D^\perp$.

The weight distributions of D and $D^\perp$ are related through the famous *MacWilliams identities*.

$$B_j(D^\perp) \quad = \quad 2^{-(n-k)} \sum_{i=0}^{n} P_j(i;n) B_i(D) \text{ for } j = 0, \ldots, n, \tag{1}$$

where $P_j(x;n) = \sum_{i=0}^{j} (-1)^i \binom{x}{i} \binom{n-x}{j-i}$ are the *Krawtchouk polynomials*.

It is easy to see from the definitions that the defining contrast subgroup of D is indeed the dual code $D^\perp$ and that the wordlength pattern of D is the weight distribution of $D^\perp$, that is,

$$A_i(D) = B_i(D^\perp) \text{ for } i = 1, \ldots, n.$$

By definition, the wordlength pattern is computed via counting words in the defining contrast subgroup. This direct approach can be cumbersome when k is large, because there are 2^k words in a 2^{n-k} design. The connection with coding theory leads to an alternative approach. We can compute $A_i(D)$ via the weight distribution $B_i(D)$ and the MacWilliams identities (1). The Krawtchouk polynomials need be computed once for each n and can be efficiently calculated via the following recursive identity:

$$P_j(x;n) = P_j(x-1;n) - P_{j-1}(x;n) - P_{j-1}(x-1;n),$$

and the initial values $P_0(x;n) = 1$ and $P_j(0;n) = \binom{n}{j}$. We use the alternative approach in our algorithm, because the alternative approach is faster than the direct approach when $k > n - k$.

2.2 Delete-one-factor projections

For a 2^{n-k} design D and $i = 1, \ldots, n$, let $D(-i)$ be the resulting $2^{(n-1)-(k-1)}$ design when the ith column is deleted. These sub-designs are called the delete-one-factor projections of D. Note that $D(-i)$ may be degenerate in the sense that it has less than 2^{n-k} distinct runs.

The next two properties about MA delete-one-factor projections are important in our construction.

Lemma 1. *For a 2^{n-k} design D, if $D(-i)$ has MA among all delete-one-factor projections of D, then the ith column is a product of some of the other columns and therefore $D(-i)$ is not degenerate.*

Proof. Suppose the result is not true, then the ith column is independent of the other columns and therefore it does not appear in any word of D. Then we can choose another column that appears in some word and deleting that column would yield a design having less aberration than $D(-i)$, which is a contradiction. □

Lemma 2. *Suppose that D is a 2^{n-k} design of resolution R with δ_n words of length R. If $D(-i)$ has MA among all delete-one-factor projections of D, then $D(-i)$ has at most $\delta_n - \lceil R \cdot \delta_n / n \rceil$ words of length R, where $\lceil x \rceil$ is the smallest integer that is greater than or equal to x.*

Proof. Each word of length R consists of R factors, so on average each factor appears in $R \cdot \delta_n / n$ words of length R. There must exist a factor that appears in at least $\lceil R \cdot \delta_n / n \rceil$ words. Deleting this factor yields a design that has at most $\delta_n - \lceil R \cdot \delta_n / n \rceil$ words of length R. The lemma follows from the fact that MA projection $D(-i)$ has the least number of words of length R. □

3 Construction Method

3.1 Basic algorithm

Following CSW, we construct designs sequentially by adding one factor at a time. We first review the basic idea of CSW's algorithm and then describe how to improve it.

Denote $r = n - k$. Let G be an $r \times (2^r - 1)$ matrix that consists of all nonzero r-tuples $(u_1, \ldots, u_r)^T$ from $GF(2)$. It is well known that every regular 2^{n-k} FF design can be viewed as n columns of an $2^r \times (2^r - 1)$ matrix H, which consists of all linear combinations of the rows of G over $GF(2)$.

Let $C^R_{n,k}$ be the set of nonisomorphic 2^{n-k} designs of resolution $\geq R$. CSW constructed $C^R_{n+1,k+1}$ from $C^R_{n,k}$ by adding an additional column. For each design in $C^R_{n,k}$, there are $2^r - 1 - n$ ways to add a column to produce a design with $n+1$ columns. Let $\tilde{C}_{n+1,k+1}$ be the set of these designs. Obviously, $|\tilde{C}_{n+1,k+1}| = (2^r - 1 - n)|C^R_{n,k}|$. It is obvious that $C^R_{n+1,k+1}$ is a subset of $\tilde{C}_{n+1,k+1}$. However, some designs in $\tilde{C}_{n+1,k+1}$ are isomorphic and some may have resolutions less than R. To construct $C^R_{n+1,k+1}$, it is necessary to eliminate these redundant designs. It is easy to eliminate designs of resolution $< R$ but is more difficult to eliminate isomorphic designs. To speed up the isomorphism check process, CSW divided all designs into different categories according to their wordlength patterns and letter patterns. Obviously, designs in different categories are not isomorphic. However, designs in the same category are not necessarily isomorphic and therefore a complete isomorphism check has to be applied to determine whether or not two designs are isomorphic.

Table 1: Number of Times that 2^{7-3} Designs are Generated in the Sequential Construction

		2^{7-3} Designs				
Design	A_3	7-3.1	7-3.2	7-3.3	7-3.4	7-3.5
6-2.1	0	**2**	**6**	0	**1**	0
6-2.2	1	0	6	**3**	0	0
6-2.3	2	0	0	9	0	0
6-2.4	2	0	2	4	1	**2**

3.2 A modified construction procedure

One problem with CSW's algorithm is that too many isomorphic designs are generated in the sequential build-up process, because a $2^{(n+1)-(k+1)}$ design can be generated from as many as $n+1$ distinct 2^{n-k} designs. We solve this problem by only allowing a design to be generated from its MA delete-one-factor projection.

We modify the construction procedure as follows. For any design D in $C_{n,k}^{R}$, adding a column to D yields a candidate design D_c. Discard D_c if its resolution is less than R or if D does not have MA among all delete-one-factor projections of D_c.

For illustration consider the construction of 2^{7-3} designs. According to CSW, there are four distinct 2^{6-2} designs and five distinct 2^{7-3} designs, labeled as 6-2.i and 7-2.j, where the designs are ranked according to the MA criterion. For each 2^{6-2} design, we can add one of the remaining 9 columns to obtain a 2^{7-2} design. Table 1 shows the number of times that each 2^{7-3} design is generated in the (unmodified) sequential construction. For example, design 7-3.3 is generated three times from design 6-2.2, nine times from design 6-2.3 and four times from design 6-2.4. The modified construction procedure only allows design 7-3.3 to be generated from design 6-2.2, because it has MA among all delete-one-factor projections of design 7-3.3. Under the original construction procedure we need entertain $4 \times 9 = 36$ designs whereas under the modified construction procedure we need entertain only 14 designs (boldfaced in Table 1). Because there are five distinct 2^{7-3} designs, we reduce the number of isomorphism checks from 31 to 9.

Bingham and Sitter (1999) proposed a construction procedure that combines the search table method of Franklin and Bailey (1977) and Franklin (1985) with the sequential approach. Table 2 shows the comparison of the construction procedures in the construction of 128-run designs of resolution ≥ 4. The last row of the table shows the number of distinct designs. As the table shows,

Table 2: Number of Designs Entertained in Creating Catalogs of 128-run Designs of Resolution ≥ 4

	n								
Procedure	8	9	10	11	12	13	14	15	16
CSW	99	458	1,104	2,597	6,632	16,200	36,192	79,064	160,040
Bingham and Sitter	99	186	506	1,367	3,499	7,950	15,798	29,062	48,889
Author	99	299	341	502	890	1,952	4,028	7,969	14,176
True	5	13	33	92	249	623	1,535	3,522	7,500

both the combined procedure of Bingham and Sitter (1999) and our modified procedure significantly reduce the number of designs entertained. For $n \geq 10$, our modified procedure entertains much less designs than the other two procedures.

We now show, by induction, that every possible 2^{n-k} design of resolution $\geq R$ in 2^r runs is isomorphic to a design in $C^R_{n,k}$ under the modified construction procedure. It is trivial that this is true for $n = r + 1$. Suppose this is true for $n = r + k$. Consider $n + 1 = r + k + 1$. Let $D = (c_1, \ldots, c_{n+1})$ be a $2^{(n+1)-(k+1)}$ design of resolution $\geq R$ in 2^r runs. Suppose that $D(-i)$ has MA among all possible delete-one-factor projections of D. Lemma 1 implies that $D(-i)$ must be a non-degenerate 2^{n-k} design of resolution $\geq R$. By the assumption for 2^{n-k} designs, there exists a design D_n in $C^R_{n,k}$ that is isomorphic to $D(-i)$. Let π be the isomorphic map from $D(-i)$ to D_n, i.e., $D_n = \pi(D(-i))$. Note that $\pi(c_i)$ is uniquely defined under this isomorphic map. Let $\pi(D) = (D_n, \pi(c_i))$. Clearly $\pi(D)$ is entertained in the modified construction procedure and therefore D is isomorphic to a design in $C^R_{n+1,k+1}$. This completes the proof.

3.3 A nonisomorphism test procedure

Xu (2005) observed that the use of wordlength patterns and letter patterns is not efficient in identifying nonisomorphic designs for three-level FF designs. Following Xu (2005), we divide designs into different categories according to their weight distributions and moment projection patterns (to be defined next). As explained in Section 2.1, the use of weight distributions is equivalent to the use of wordlength patterns in terms of distinguishing designs but is more efficient in terms of computation (when $k > r$).

For a 2^{n-k} design D and an integer p, $p < n$, there are $\binom{n}{p}$ p-factor projections. For each

Table 3: Number of Designs Identified for 128-Run Designs of Resolution ≥ 4

	n								
Method	8	9	10	11	12	13	14	15	16
WLP	5	13	28	68	152	297	518	889	1,425
LP	5	13	33	92	247	617	1,506	3,467	7,229
MPP ($q = 1$)	5	13	33	92	247	617	1,506	3,467	7,229
MPP ($q = 2$)	5	13	33	92	249	623	1,535	3,522	7,500

p-factor projection, say D_p, and an integer t, compute the tth power moment

$$K_t(D_p) = \sum_{i=0}^{p}(p-i)^t B_i(D_p),$$

where $B_i(D_p)$ is the number of row vectors of D_p with Hamming weight i. The power moment K_t was introduced by Xu (2003) and Xu and Deng (2005) for ranking and classifying nonregular designs. The frequency distribution of K_t-values of all p-factor projections is called the *p-dimensional K_t-value distribution*. It is evident that isomorphic designs have the same p-dimenionsonal K_t-value distribution for all positive integers t and $p < n$. Whenever two designs have different p-dimensional K_t-value distributions for some t and p, these two designs must be nonisomorphic.

To ease the computation, we fix t and let p vary from $n-1$ to $n-q$, where q is a pre-chosen small number, say 2 or 3. The corresponding q K_t-value distributions are called the moment projection pattern. It requires $O(n^q)$ operations to compute the moment projection pattern. The choice of t does not make a difference provided $t > 5$ in most cases. In the algorithm, we fix t arbitrarily at $t = 10$.

Table 3 shows the numbers of designs identified by the wordlength pattern (WLP), letter pattern (LP), moment projection pattern (MPP) with $q = 1$ and 2 in the construction of 128-run designs of resolution ≥ 4 for $n \leq 16$. Note that the moment projection pattern with $q = 1$ and the letter pattern identify the same numbers of designs. The moment projection pattern with $q = 2$ correctly identifies all nonisomorphic designs for $n \leq 16$.

As Table 3 shows, the moment projection pattern check with $q = 1$ has the same or nearly the same classification power as the letter patten check whereas the moment projection pattern check with $q = 2$ or 3 typically has more classification power. Furthermore, when k is large, the moment projection pattern check is faster than the letter pattern check.

3.4 A fast isomorphism check procedure

We first review the isomorphism check procedure proposed by CSW. Consider two 2^{7-3} designs defined by

$$D_1 : 5 = 123, 6 = 124, 7 = 13 \text{ and } D_2 : 5 = 12, 6 = 124, 7 = 234,$$

which have the same wordlength pattern and letter pattern. CSW's procedure works as follows:

1. Select four independent columns from D_2, say, $\{1, 2, 3, 6\}$. There are $\binom{7}{4}$ choices.
2. Select a relabeling map from $\{1, 2, 3, 6\}$ to $\{a, b, c, d\}$, say, $a = 1$, $b = 2$, $c = 3$, and $d = 6$. There are 4! choices.
3. Write the remaining columns, $\{4, 5, 7\}$, in D_2 as interactions of $\{a, b, c, d\}$, i.e., $4 = abd$, $5 = ab$, and $7 = acd$. Then D_2 can be written as $\{a, b, c, d, ab, abd, acd\}$.
4. Compare the new representation of D_2 with that of D_1. If they match, D_1 and D_2 are isomorphic, and the process stops. Otherwise, return to step 2 and try another map of $\{a, b, c, d\}$. When all the relabeling maps are exhausted, return to step 1 and find next four columns.

If two designs are isomorphic, an isomorphic map will be found eventually. Otherwise, two designs are not isomorphic. In the worst case, it requires $O(n\binom{n}{r}r!)$ operations to declare that two 2^{n-k} designs are not isomorphic.

We improve the isomorphism check procedure by considering delete-one-factor projections. Let π be a permutation of $\{1, \ldots, n\}$. If π is an isomorphic map from D_1 to D_2, $D_1(-i)$ and $D_2(-\pi(i))$ must be isomorphic and therefore they must have the same weight distribution. So π cannot be an isomorphic map if $D_1(-i)$ and $D_2(-\pi(i))$ do not have the same weight distribution for some i.

For convenience, we call a permutation π *feasible* if $D_1(-i)$ and $D_2(-\pi(i))$ have the same weight distribution for every i. A relabeling map is feasible if its induced permutation is feasible. The key idea of our new isomorphism check procedure is to entertain only feasible relabeling maps by matching the factors using the weight distributions of the delete-one-factor projections.

We illustrate our procedure with the two 2^{7-3} designs mentioned earlier. Here are the steps.

1. Compute the weight distributions of the delete-one-factor projections (delete-one weight distributions, for short) for both designs; see Table 4. For each column of D_1, count the frequency that each delete-one weight distribution appears. Let n_i be the frequency for the ith column. Here $n_1 = n_2 = n_3 = n_5 = 4$, $n_4 = n_6 = 2$ and $n_7 = 1$.

Table 4: Weight Distributions of Delete-One-Factor Projections

D_1								D_2							
Projection	B_0	B_1	B_2	B_3	B_4	B_5	B_6	Projection	B_0	B_1	B_2	B_3	B_4	B_5	B_6
$D_1(-1)$	1	0	4	6	3	2	0	$D_2(-1)$	1	0	4	6	3	2	0
$D_1(-2)$	1	0	4	6	3	2	0	$D_2(-2)$	1	0	4	6	3	2	0
$D_1(-3)$	1	0	4	6	3	2	0	$D_2(-3)$	1	1	2	6	5	1	0
$D_1(-4)$	1	1	2	6	5	1	0	$D_2(-4)$	1	0	4	6	3	2	0
$D_1(-5)$	1	0	4	6	3	2	0	$D_2(-5)$	1	0	3	8	3	0	1
$D_1(-6)$	1	1	2	6	5	1	0	$D_2(-6)$	1	0	4	6	3	2	0
$D_1(-7)$	1	0	3	8	3	0	1	$D_2(-7)$	1	1	2	6	5	1	0

2. Relabel the columns of D_1 by selecting four new independent columns so that their frequency numbers n_i are as small as possible. For example, we select columns $\{7, 4, 6, 1\}$ as the new independent columns. We relabel them as $\{a, b, c, d\}$, i.e., $a = 7, b = 4, c = 6, d = 1$, and write the remaining three columns as their interactions, i.e., $2 = bcd$, $3 = ad$, and $5 = abcd$. So after relabeling, D_1 becomes $D_1' : \{a, b, c, d, ad, bcd, abcd\}$. The purpose of this step is to reduce the number of feasible relabeling maps to be considered in the next step.

3. Select four independent columns from D_2 that have the same delete-one weight distributions as the four independent columns from D_1', and relabel the columns. To obtain a feasible map from D_2 to D_1', we must relabel column 5 of D_2 as a, because only column 5 has the same delete-one weight distribution as factor a of D_1'. Similarly, we must relabel column 3 or 7 of D_2 as b or c. We can relabel column 1, 2, 4, or 6 of D_2 as d. There are $1 \times 2 \times 4 = 8$ choices of feasible relabeling maps. For example, we choose $a = 5, b = 3, c = 7, d = 1$ and write the remaining columns as $2 = ad, 4 = abcd, 6 = bcd$. It is clear now that D_2 is isomorphic to D_1' and hence to D_1.

4. If two designs do not match after relabeling the independent columns, consider another choice of relabeling and/or another choice of independent columns in step 3. If none of the choices yields to an identical design, two designs are not isomorphic.

In the above example, we entertain only eight feasible relabeling maps out of $\binom{7}{4}4! = 840$ possible choices of relabeling maps. It can be verified that any of the eight feasible relabeling maps leads to an isomorphic map. This is not true in general.

Table 5: Time to Create Catalogs of 128-Run Designs of Resolution ≥ 4

	n								
Algorithm	8	9	10	11	12	13	14	15	16
CSW	0s	1s	4s	27s	2m32s	10m30s	37m48s	2h27m	6h43m
Author	0s	0s	0s	1s	1s	4s	8s	16s	39s

NOTE: The CSW's algorithm is modified so that two algorithms differ only in the isomorphism check procedures used. The h, m, and s stand for hour, minute, and second, respectively.

In theory our new isomorphism check procedure still requires $O(n\binom{n}{r}r!)$ operations in the worst case. In practice, the new isomorphism check procedure saves tremendous amount of computer time, because the worst case happens rarely.

To see the computation advantage of the our new isomorphism check procedure, we develop two algorithms with everything the same except isomorphism check procedures, one with the original procedure by CSW and the other with our new procedure. Table 5 shows the real time comparison of these two procedures in constructing 128-run designs of resolution ≥ 4. The savings are tremendous and become larger for larger designs. The times are taken on a 2GHz PowerPC G5 computer.

The isomorphism check can be made faster in some situations. It is evident that two designs are isomorphic if and only if their dual codes are isomorphic. So when $k < r$, we perform isomorphism checks on the dual codes. This technique was previously used by Lin and Sitter (2006).

As an alternative, we can match columns using their letter patterns. It can be shown that the use of delete-one weight distributions is equivalent to the use of letter patterns. We use the former because it is faster to compute delete-one weight distributions than letter patterns when $k > r$.

Clark and Dean (2001) presented a method of determining isomorphism of any two FF designs, regular or nonregular, by examining the Hamming distances of their projection designs. They also developed an algorithm for checking the isomorphism of two-level designs. Their isomorphism check procedure, adopted by Lin and Sitter (2006), is inferior to ours for the regular design case, because it ignores the special property of regular designs and requires $O(n(n!)^2)$ operations in theory for the worst case.

Table 6: Illustration of Constructing MA 256-Run Designs for $n \leq 28$

n	9	10	11	12	13	14	15	16	17	18
δ_n	0	0	0	0	0	1	2	3	5	7
$\lvert C^4_{n,k}(\delta_n)\rvert$	5	9	11	14	15	124	617	1,836	14,158	46,929
n	19	20	21	22	23	24	25	26	27	28
δ_n	9	12	16	20	25	31	38	46	54	64
$\lvert C^4_{n,k}(\delta_n)\rvert$	56,821	104,654	258,535	136,105	65,070	23,981	5,610	661	6	1

3.5 Construction of MA designs

It is impractical to enumerate all designs in many situations. Here we propose a method for constructing MA designs by enumerating a subset of good designs.

Let $C^R_{n,k}(\delta_n)$ be the set of nonisomorphic 2^{n-k} designs of resolution $\geq R$ with at most δ_n words of length R. We can sequentially build up $C^R_{n,k}(\delta_n)$ as before. To construct $C^R_{n,k}(\delta_n)$, according to Lemma 2, it is sufficient to add a column to every design in $C^R_{n-1,k-1}(\delta_{n-1})$, where

$$\delta_{n-1} = \delta_n - \left\lceil \frac{R \cdot \delta_n}{n} \right\rceil . \tag{2}$$

For illustration, consider the construction of MA 256-run designs for $n \leq 28$. It is known from Block (2003) that there is a resolution four 2^{28-20} design with $A_4 = 64$. We set $R = 4$, $\delta_{28} = 64$, and compute δ_{n-1} backward using (2) recursively for $n = 28, \ldots, 10$. Then we build up $C^4_{n,k}(\delta_n)$ forward for $n = 9, \ldots, 28$. By completely enumerating $C^4_{n,k}(\delta_n)$, we obtain all MA 256-run designs for $n \leq 28$. Table 6 shows the value of δ_n and the cardinality of $C^4_{n,k}(\delta_n)$. From the table, we know that there is a unique resolution four 2^{28-20} design with $A_4 \leq 64$.

3.6 Construction of even and odd designs

Sometimes it is of interest to study even and odd designs separately. Recall that an even design contains entirely defining words of even length whereas an odd design has at least one defining word of odd length. It is evident that any projection of an even design is an even design. So when searching for even designs, we can simply discard all odd designs. On the other hand, a projection of an odd design can be an odd or even design, and at least one projection is an odd design. So when searching for odd designs, we add an extra step in our construction procedure.

Here is the modified construction procedure for odd designs. Let $O_{n,k}^R$ be the set of nonisomorphic odd 2^{n-k} designs of resolution $\geq R$. For any design D in $O_{n,k}^R$, adding a column to D yields a candidate design D_c. Discard D_c if its resolution is less than R or if it is an even design. Further discard D_c if all of its delete-one-factor projections are odd designs and D does not have MA among all the delete-one-factor projections. Let $\tilde{O}_{n+1,k+1}^R$ be the set of designs D_c left. It can be shown that $\tilde{O}_{n+1,k+1}^R$ contains $O_{n+1,k+1}^R$.

Whether a design is even or odd can be conveniently determined by its weight distribution.

Lemma 3. *(a) A 2^{n-k} design D is an odd design if and only if $B_n(D) = 0$.*

(b) For an odd 2^{n-k} design D, all of its delete-one-factor projections are odd designs if and only if $B_{n-1}(D) = 0$.

Proof. (a) It is well known that a 2^{n-k} design is an even design if and only if it is a fold-over design. Because every 2^{n-k} design contains a row of 0's, an even design contains a row of 1's. So D is an even design if and only if $B_n(D) = 1$. Equivalently, D is an odd design if and only if $B_n(D) = 0$.

(b) It follows from (a). □

4 Tables of Designs

Using the new algorithm we completely enumerate all 128-run designs of resolution ≥ 4 up to 30 factors, all 256-run designs of resolution ≥ 4 up to 17 factors, all 512-run designs of resolution ≥ 5, all 1024-run designs of resolution ≥ 6, and all 2048- and 4096-run designs of resolution ≥ 7. Table 7 shows the number of nonisomorphic designs for various run sizes and resolutions.

By constructing even and odd designs separately, we further completely enumerate all odd 128-run designs of resolution ≥ 4 which exist for $n \leq 40$ and all even 256-run designs of resolution ≥ 4 for $n \leq 19$. Table 8 shows the number of nonisomorphic even and odd designs for 128, 256, 512, and 1024 runs. The complete set of designs can be obtained from the author upon request.

The 128-run designs are of special interest because MA designs are given in CSW up to 64 runs. Block and Mee (2005) constructed MA and weak MA 128-run designs for n=12–64. They achieved this by enumerating all odd designs of resolution four and all even designs for $n \leq 22$, based on their conjecture. By comparing the numbers of even and odd designs, we conclude that their set of odd designs is complete and their set of even designs is also complete for $n \leq 22$. The numbers of even designs for $n = 21$ and 22 in their table 6 are not correct, though. So their conjecture is correct for all the cases they considered and their designs do have weak MA as claimed except for a few typos in their table 2 with 15, 19–21, and 30–32 factors (see Corrigenda). For easy reference, we give all MA and weak MA designs for 128 runs up to 40 factors in Table 9, constructed according to the procedure in Section 3.5. Note from Table 9 that MA designs are in sequential order for n=32–40. However, this is not true for $n = 31$, which agrees with the theoretical result of Xu and Cheng (2007). For $40 < n \leq 64$, MA designs can be obtained via deleting the MA complementary even designs from the unique even 2^{64-57} design; see Butler (2003) and Block and Mee (2005) for details. Again, this can be achieved by enumerating a set of good even designs. We confirm that MA designs are unique except for n=41, 42, 43, 44, and 50. For $n > 64$, MA designs can also be obtained via complementary designs; see Chen and Hedayat (1996), Tang and Wu (1996), Butler (2003), and Xu and Cheng (2007). Thus, all MA 128-run designs can be constructed.

Table 10 gives all MA and weak MA 256-run designs up to 28 factors, constructed as outlined in Section 3.5. Block (2003) previously studied 256-run designs and obtained a list of designs up to 80 factors. Comparing the wordlength patterns, we find that his designs do not have MA when $n = 24, 25$.

Tables 11 and 12 give selected 512-run designs of resolution ≥ 5 and 1024-run designs of resolution ≥ 6. These tables include all weak MA designs and the top three or four designs. We list only one design when other designs have less resolution. MA designs are unique and weak MA designs are also unique except for 512 runs and $n = 22$.

Draper and Mitchell (1970) conjectured that all 2^{23-14} designs of resolution five are equivalent. We confirm this; see Table 7. According to Table 8, there are 5,710 nonisomorphic even 1024-run designs of resolution ≥ 6. Draper and Mitchell (1970) identified 4,043 even designs using the letter pattern check, so they missed 1,667 (about 30%) even designs.

Table 13 gives the complete catalog of 2048-run designs of resolution ≥ 7 and Table 14 gives the complete catalog of 4096-run designs of resolution ≥ 8. Weak MA designs are unique for all cases. According to Table 7, 4096-run designs of both resolution 7 and 8 have maximum 24 factors.

In Tables 9–14, each 2^{n-k} design is labeled as $n - k.i$, where the index i reflects the ordering based on the MA criterion. When two or more designs have the same wordlength pattern, the ordering among them is arbitrary. Every 2^{n-k} design is represented by a set of n columns in the Yates order. To save space, we omit the independent columns, which are $\{1, 2, \ldots, 2^{n-k-1}\}$, and give only a set of k columns. For illustration, consider design 9-2.1 in Table 9 which has columns {31, 103}. Denote the nine factors as $\{x_1, \ldots, x_9\}$, where $\{x_1, \ldots, x_7\}$ represent independent columns, that is, $x_i = 2^{i-1}$ for $i = 1, \ldots, 7$. Then $x_8 = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5$ and $x_9 = x_1 \cdot x_2 \cdot x_3 \cdot x_6 \cdot x_7$ because

$31 = 2^0 + 2^1 + 2^2 + 2^3 + 2^4$ and $103 = 2^0 + 2^1 + 2^2 + 2^5 + 2^6$. The wordlength pattern of this design is $A_6 = 3$ and $A_i = 0$ for $i \neq 6$.

5 Concluding Remarks

We develop a new sequential algorithm for constructing large FF designs. The new algorithm has the following features:

1. A construction procedure that allows a design to be constructed only from its MA projection in the sequential build-up process,

2. A nonisomorphism test procedure that uses moment projection patterns to identify nonisomorphic designs efficiently,

3. A fast isomorphism check procedure that matches factors using their delete-one weight distributions,

4. A method for constructing MA designs using a partial catalog of good designs.

With some proper modifications, these features can be used to more efficiently construct designs for other situations such as blocked designs, split-plot designs, and robust parameter designs.

Using this algorithm we construct MA designs with run sizes ranging from 128 to 4096 and maximum number of factors ranging from 23 to 64. The construction of large designs beyond these presented in this paper remains challenging, because of the enormous number of designs encountered, as evidenced in Tables 7 and 8. It appears feasible to enumerate all resolution four designs with 128 runs on a personal computer; however, it seems impractical to do so with 256 runs at this moment. The algorithm takes one hour on a G5 computer to enumerate all resolution four designs with 256 runs for $n = 16$ and one day for $n = 17$, and in the latter case there are about 1.8 million distinct designs. Further research is called for.

Acknowledgments

The research was supported by National Science Foundation grant DMS-0505728. The author thanks Robert Mee and Robert Block for sharing their designs and helpful communications, and Randy Sitter for sharing their manuscript.

REFERENCES

Bingham, D., and Sitter, R. R. (1999), "Minimum Aberration Two-Level Fractional Factorial Split-Plot Designs," *Technometrics*, 41, 62–70.

Block, R. M. (2003), *Theory and Construction Methods for Large Regular Resolution IV Designs*, Ph.D. dissertation, University of Tennessee, Knoxville.

Block, R. M., and Mee, R. W. (2005), "Resolution IV Designs With 128 Runs," *Journal of Quality Technology*, 37, 282–293. Corrigenda, (2006), 38, 196.

Box, G. E. P., and Hunter, J. S. (1961), "The 2^{k-p} Fractional Factorial Designs," *Technometrics*, 3, 311–351, 449–458.

Box, G. E. P., Hunter, W. G., and Hunter, J. S. (2005), *Statistics for Experimenters*, 2nd ed., New York: Wiley.

Butler, N. A. (2003), "Some Theory for Constructing Minimum Aberration Fractional Factorial Designs," *Biometrika*, 90, 233–238.

Chen, C. L. (1991), "Construction of Some Binary Linear Codes of Minimum Distance Five," *IEEE Transactions on Information Theory*, 37, 1429–1432.

Chen, H., and Hedayat, A. S. (1996), "2^{n-l} Designs With Weak Minimum Aberration," *The Annals of Statistics*, 24, 2536–2548.

Chen, J., and Lin, D. K. J. (1991), "On the Identity Relationship of 2^{k-p} Designs," *Journal of Statistical Planning and Inference*, 28, 95–98.

Chen, J., Sun, D. X., and Wu, C. F. J. (1993), "A Catalogue of Two-Level and Three-Level Fractional Factorial Designs With Small Runs," *International Statistical Review*, 61, 131–145.

Clark, J. B., and Dean, A. M. (2001), "Equivalence of Fractional Factorial Designs," *Statistica Sinica*, 11, 537–547.

Dean, A. M., and Voss, D. T. (1999), *Design and Analysis of Experiments*, New York: Springer.

Draper, N. R., and Mitchell, T. J. (1967), "The Construction of Saturated 2_{V}^{k-p} Designs," *The Annals of Mathematical Statistics*, 38, 1110–1126.

Draper, N. R., and Mitchell, T. J. (1968), "Construction of the Set of 256-Run Designs of Resolution ≥ 5 and the Set of Even 512-Run Designs of Resolution ≥ 6 With Special Reference to the Unique Saturated Designs," *The Annals of Mathematical Statistics*, 39, 246–255.

Draper, N. R., and Mitchell, T. J. (1970), "Construction of a Set of 512-Run Designs of Resolution ≥ 5 and a Set of Even 1024-Run Designs of Resolution ≥ 6," *The Annals of Mathematical Statistics*, 41, 876–887.

Franklin, M. F. (1985), "Selecting Defining Contrasts and Confounded Effects in p^{n-m} Factorial Experiments," *Technometrics*, 27, 165–172.

Franklin, M. F., and Bailey, R. A. (1977), "Selection of Defining Contrasts and Confounded Effects in Two-Level Experiments," *Applied Statistics*, 26, 321–326.

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

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

Lin, C. D., and Sitter, R. R. (2006), "Isomorphism of Fractional Factorial Designs," preprint.

Ma, C.-X., Fang, K.-T., and Lin, D. K. J. (2001). "On the Isomorphism of Fractional Factorial Designs," *Journal of Complexity*, 17, 86–97.

MacWilliams, F. J., and Sloane, N. J. A. (1977), *The Theory of Error-Correcting Codes*, Amsterdam: North-Holland.

Mee, R. W. (2004), "Efficient Two-Level Designs for Estimating Main Effects and Two-Factor Interactions," *Journal of Quality Technology*, 36, 400–412.

Montgomery, D. C. (2005), *Design and Analysis of Experiments*, 6th ed., New York: Wiley.

Mukerjee, R., and Wu, C. F. J. (2006), *A Modern Theory of Factorial Designs*, New York: Springer.

Tang, B., and Wu, C. F. J. (1996), "Characterization of Minimum Aberration 2^{n-m} Designs in Terms of Their Complementary Designs," *The Annals of Statistics*, 24, 2549–2559.

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

Xu, H. (2003), "Minimum Moment Aberration for Nonregular Designs and Supersaturated Designs," *Statistica Sinica*, 13, 691–708.

Xu, H. (2005), "A Catalogue of Three-Level Regular Fractional Factorial Designs," *Metrika*, 62, 259–281.

Xu, H., and Cheng, C. -S. (2007), "A Complementary Design Theory for Doubling," *The Annals of Statistics*, to appear.

Xu, H., and Deng, L. -Y. (2005), "Moment Aberration Projection for Nonregular Fractional Factorial Designs," *Technometrics*, 47, 121–131.

Zhu, Y., and Zeng, P. (2005), "On the Coset Pattern Matrices and Minimum M-Aberration of 2^{n-p} Designs," *Statistica Sinica*, 15, 717–730.

Table 7: Number of Nonisomorphic Designs

	Run Size (Resolution $\geq R$)							
n	$2^7(4)$	$2^8(4)$	$2^8(5)$	$2^9(5)$	$2^{10}(6)$	$2^{11}(7)$	$2^{12}(7)$	$2^{12}(8)$
8	5							
9	13	6	5					
10	33	21	9	6				
11	92	74	11	16	6			
12	249	311	14	36	14	6		
13	623	1,429	15	92	24	9	7	6
14	1,535	7,344	11	282	47	7	17	7
15	3,522	42,581	6	1,011	98	7	27	4
16	7,500	271,784	1	4,019	185	7	48	5
17	14,438	1,798,534	1	13,759	380	3	95	5
18	25,064	?	0	29,373	919	2	113	2
19	39,335	?		31,237	1,701	1	84	1
20	57,920	?		14,135	1,682	1	35	1
21	82,496	?		2,373	739	1	22	1
22	118,444	?		128	128	1	17	1
23	173,092	?		1	8	1	17	1
24	256,654	?		0	1	0	13	1
25	376,382	?			0		0	0
26	537,907	?						
27	735,111	?						
28	956,190	?						
29	1,174,404	?						
30	1,363,003	?						
Total	?	?	73	96,468	5,932	46	495	35

Table 8: Number of Nonisomorphic Even and Odd Designs

	Run Size (Resolution ≥ R)							
	$2^7(4)$		$2^8(4)$		$2^9(5)$		$2^{10}(6)$	
n	Even	Odd	Even	Odd	Even	Odd	Even	Odd
8	3	2						
9	6	7	3	3				
10	14	19	9	12	3	3		
11	30	62	24	50	4	12	3	3
12	69	180	80	231	5	31	7	7
13	136	487	241	1,188	5	87	11	13
14	295	1,240	839	6,505	5	277	23	24
15	596	2,926	3,029	39,552	5	1,006	51	47
16	1,292	6,208	12,487	259,297	3	4,016	125	60
17	2,651	11,787	55,331	1,743,203	1	13,758	332	48
18	5,598	19,466	265,798	?	1	29,372	908	11
19	11,341	27,994	1,314,705	?	0	31,237	1,695	6
20	22,728	35,192	?	?		14,135	1,681	1
21	43,295	39,201	?	?		2,373	738	1
22	79,597	38,847	?	?		128	127	1
23	138,224	34,868	?	?		1	8	0
24	228,521	28,133	?	?		0	1	
25	355,813	20,569	?	?			0	
26	524,409	13,498	?	?				
27	727,036	8,075	?	?				
28	951,906	4,284	?	?				
29	1,172,255	2,149	?	?				
30	1,362,027	976	?	?				
31	?	433	?	?				
32	?	197	?	?				
33	?	101	?	?				
34	?	31	?	?				
35	?	13	?	?				
36	?	8	?	?				
37	?	3	?	?				
38	?	2	?	?				
39	?	1	?	?				
40	?	1	?	?				
Total	?	296,960	?	?	32	96,436	5,710	222

Table 9: Weak Minimum Aberration 128-Run Designs for $n \leq 40$

Design	$(A_4, A_5, \ldots)$	Columns
8-1.1	0 0 0 0 1	127
9-2.1	0 0 3 0 0	31 103
10-3.1	0 3 3 1 0	31 103 43
11-4.1	0 6 6 2 1	31 103 43 85
12-5.1	1 8 12 8 1	31 103 43 85 121
12-5.2	1 10 10 5 4	31 103 43 85 44
12-5.3	1 10 11 4 2	31 103 43 85 46
13-6.1	2 16 18 10 9	31 103 43 85 44 86
13-6.2	2 16 20 8 5	31 103 43 85 46 61
14-7.1	3 24 36 16 11	31 103 43 85 46 61 114
15-8.1	7 32 52 40 35	31 103 43 85 46 61 114 67
15-8.2	7 34 46 42 45	31 103 43 85 44 86 88 53
15-8.3	7 38 44 28 51	31 103 43 85 46 61 114 13
16-9.1	10 48 72 80 90	31 103 43 85 44 86 88 53 110
17-10.1	15 60 130 120	31 103 43 85 46 61 114 67 78 116
17-10.2	15 66 110 130	31 103 43 85 46 61 114 67 78 55
17-10.3	15 68 106 128	31 103 43 85 44 86 88 53 38 58
17-10.4	15 72 102 112	31 103 43 85 46 61 114 67 13 55
18-11.1	20 80 200 192	31 103 43 85 46 61 114 67 78 116 121
18-11.2	20 92 160 212	31 103 43 85 46 61 114 67 78 55 58
19-12.1	27 120 235 344	31 103 43 85 46 61 114 67 78 55 58 86
20-13.1	36 152 340 544	31 103 43 85 46 61 114 67 78 55 58 86 91
21-14.1	51 200 414 840	31 103 43 85 44 82 54 56 88 78 123 125 104 25
21-14.2	51 202 400 860	31 103 43 85 44 86 88 53 38 58 79 83 110 124
22-15.1	65 248 572 1280	31 103 43 85 44 86 88 53 78 58 83 97 28 104 114
22-15.2	65 256 552 1256	31 103 43 85 44 82 54 56 88 78 123 125 104 25 112
23-16.1	83 316 744	31 103 43 85 44 82 54 56 88 78 123 125 104 25 112 49
23-16.2	83 318 734	31 103 43 85 44 86 88 53 38 58 79 83 110 124 97 104
24-17.1	102 384 992	31 103 43 85 44 86 88 53 110 19 28 57 67 98 100 26 105
24-17.2	102 394 985	31 103 43 85 44 86 88 53 38 58 79 83 110 124 97 104 114
25-18.1	124 482 1312	31 103 43 85 44 86 88 53 38 58 79 83 110 124 97 104 114 123
26-19.1	152 568 1704	31 103 43 85 44 86 88 53 110 19 28 57 67 98 100 26 105 62 77
27-20.1	180 690 2200	31 103 43 85 44 86 88 53 110 19 28 57 67 98 100 26 105 62 77 112
28-21.1	210 840 2800	31 103 43 85 44 86 88 53 110 19 28 57 67 98 100 26 105 62 77 112 127
29-22.1	266 945 3472	31 103 43 85 44 86 88 53 110 19 28 57 67 98 100 26 105 62 77 112 127 124
30-23.1	335 972 4662	31 103 43 81 45 26 114 127 22 67 56 94 116 7 38 108 14 69 53 25 73 121 28
31-24.1	391 1134 5826	same as design 30-23.1, plus 91
31-24.2	391 1134 5827	same as design 30-23.1, plus 51
32-25.1	452 1322 7219	same as design 30-23.1, plus 51 97
32-25.2	452 1323 7218	same as design 30-23.1, plus 91 51
32-25.3	452 1324 7219	same as design 30-23.1, plus 51 62
33-26.1	518 1543 8863	same as design 30-23.1, plus 51 97 70
33-26.2	518 1544 8863	same as design 30-23.1, plus 91 51 62
34-27.1	589 1800 10788	same as design 30-23.1, plus 51 97 70 79
34-27.2	589 1801 10788	same as design 30-23.1, plus 51 97 70 87
35-28.1	665 2100 13020	same as design 30-23.1, plus 51 97 70 79 93
35-28.2	665 2101 13020	same as design 30-23.1, plus 51 97 70 79 91
36-29.1	756 2401 15736	same as design 30-23.1, plus 51 97 70 79 93 62
37-30.1	854 2744 18886	same as design 30-23.1, plus 51 97 70 79 93 62 87
38-31.1	959 3136 22512	same as design 30-23.1, plus 51 97 70 79 93 62 87 88
39-32.1	1071 3584 26656	same as design 30-23.1, plus 51 97 70 79 93 62 87 88 91
40-33.1	1190 4096 31360	same as design 30-23.1, plus 51 97 70 79 93 62 87 88 91 106

Table 10: Weak Minimum Aberration 256-Run Designs for $n \leq 28$

Design	$(A_4, A_5, \ldots)$	Columns
9-1.1	0 0 0 0 0 1	255
10-2.1	0 0 1 2 0 0	63 199
11-3.1	0 0 6 0 1 0	127 143 179
12-4.1	0 0 12 0 3 0	127 143 179 213
13-5.1	0 3 12 12 3 0	127 143 179 213 105
14-6.1	0 9 18 16 7 6	127 143 179 213 105 27
15-7.1	0 15 30 26 15	127 143 179 213 105 27 46
16-8.1	0 24 44 40 45	127 143 179 85 150 75 108 189
17-9.1	0 34 68 68 85	127 143 179 85 150 75 108 189 229
18-10.1	3 36 114 132	127 143 179 213 105 27 46 182 92 194
18-10.2	3 42 102 118	127 143 179 213 105 27 46 152 203 214
18-10.3	3 44 98 116	127 143 179 213 105 27 46 92 152 203
18-10.4	3 46 92 111	127 143 179 85 150 75 108 189 229 7
18-10.5	3 47 95 103	127 143 179 213 105 27 46 77 158 185
18-10.6	3 48 94 100	127 143 179 213 105 27 46 77 185 234
18-10.7	3 48 96 100	127 143 179 85 150 75 108 118 184 234
18-10.8	3 48 102 92	127 143 179 213 105 27 46 158 164 185
19-11.1	4 48 168 208	127 143 179 213 105 27 46 182 92 194 229
19-11.2	4 56 152 184	127 143 179 213 105 27 46 152 203 214 236
19-11.3	4 60 144 180	127 143 179 213 105 27 46 152 203 214 92
19-11.4	4 64 140 160	127 143 179 213 105 27 46 77 158 185 234
19-11.5	4 64 152 144	127 143 179 213 105 27 46 158 164 185 234
20-12.1	5 64 240 320	127 143 179 213 105 27 46 182 92 194 229 248
20-12.2	5 80 208 272	127 143 179 213 105 27 46 152 203 214 236 92
21-13.1	9 104 268	127 143 179 213 105 27 46 152 203 214 236 92 45
22-14.1	14 137 346	127 143 179 213 105 27 46 77 158 185 234 164 88 201
23-15.1	20 172 450	127 143 179 213 105 27 46 77 158 185 234 201 88 43 236
24-16.1	26 216 584	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165
25-17.1	34 262 760	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 49
25-17.2	34 266 752	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 78
26-18.1	43 325 963	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 78 113
26-18.2	43 326 960	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 78 124
27-19.1	53 395 1224	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 78 113 124
28-20.1	64 476 1550	127 143 179 213 105 27 46 77 158 185 84 248 166 83 146 165 78 113 124 228

Table 11: Selected 512-Run Designs of Resolution ≥ 5

Design	$(A_5, A_6, \ldots)$	Columns
10-1.1	0 0 0 0 0 1	511
11-2.1	0 0 2 1 0 0	127 399
12-3.1	0 2 4 1 0 0	127 399 179
12-3.2	0 3 3 0 1 0	255 271 307
12-3.3	0 4 0 3 0 0	127 391 155
12-3.4	0 4 0 3 0 0	127 143 307
13-4.1	0 4 8 3 0 0	127 399 179 341
13-4.2	0 6 6 1 2 0	127 399 179 213
13-4.3	0 8 0 7 0 0	127 391 155 301
14-5.1	0 7 16 7 0 0	127 399 179 341 489
14-5.2	0 15 0 14 0 1	127 391 155 301 206
14-5.3	0 15 0 15 0 0	127 391 155 301 433
15-6.1	0 25 0 30 0 3	127 391 155 301 206 501
15-6.2	0 27 0 23 0 12	127 391 155 301 433 205
15-6.3	0 27 0 24 0 9	127 391 155 301 206 188
16-7.1	0 44 0 45 0 28	127 391 155 301 206 188 358
16-7.2	0 45 0 41 0 34	127 391 155 301 433 205 345
16-7.3	0 48 0 30 0 48	127 143 307 181 211 285 327
17-8.1	0 68 0 85 0 68	127 391 155 301 206 188 358 369
18-9.1	0 102 0 153 0	127 391 155 301 206 188 358 369 468
19-10.1	12 84 156 78	127 143 307 181 211 285 327 105 427 473
19-10.2	15 102 67 153	127 391 155 301 206 188 358 369 468 15
19-10.3	18 78 130 104	127 143 307 181 211 285 105 427 473 39
20-11.1	16 120 240 130	127 143 307 181 211 285 327 105 427 473 485
20-11.2	24 112 200 170	127 143 307 181 211 285 327 105 427 473 39
20-11.3	28 108 188 182	127 143 307 181 211 285 105 427 473 39 60
21-12.1	21 168 360 210	127 143 307 181 211 285 327 105 427 473 485 510
21-12.2	37 152 280 290	127 143 307 181 211 285 327 105 427 473 39 60
21-12.3	45 139 240 354	127 143 307 181 211 285 105 427 473 39 323 453
22-13.1	63 189 325 569	127 391 155 301 206 188 358 350 507 105 298 275 369
22-13.2	63 190 321 566	127 391 155 301 206 188 358 369 83 166 197 408 498
22-13.3	63 190 321 566	127 391 155 301 206 188 358 23 340 507 418 99 46
22-13.4	63 193 313 565	127 391 155 301 206 188 358 369 468 15 86 99 166
23-14.1	84 252 445 890	127 391 155 301 206 188 358 23 340 430 435 90 450 99

Table 12: Selected 1024-Run Designs of Resolution ≥ 6

Design	$(A_6, A_7, \ldots)$	Columns
11-1.1	0 0 0 0 0 1	1023
12-2.1	0 0 3 0 0 0	127 911
13-3.1	0 4 3 0 0 0	127 911 435
14-4.1	0 8 7 0 0 0	127 911 435 725
15-5.1	0 15 15 0 0	127 911 435 725 873
16-6.1	6 25 15 0 10	127 911 435 725 873 158
16-6.2	8 24 13 0 8	127 911 435 725 158 327
16-6.3	9 16 18 12 3	127 911 435 213 665 865
17-7.1	12 41 25 0 20	127 911 435 725 873 158 327
17-7.2	13 40 25 0 18	127 911 435 725 158 327 490
17-7.3	15 34 23 14 16	127 911 435 725 158 327 551
18-8.1	19 66 45 0 42	127 911 435 725 873 158 327 490
18-8.2	20 64 46 0 40	127 911 435 725 158 327 490 860
18-8.3	28 48 38 32 32	127 911 435 469 158 376 535 1022
19-9.1	28 104 78 0 88	127 911 435 725 873 158 327 490 626
19-9.2	46 56 81 72 81	127 911 435 213 665 301 625 598 841
19-9.3	48 54 81 72 72	127 911 435 213 665 301 625 598 762
20-10.1	40 160 130 0	127 911 435 725 873 158 327 490 626 697
20-10.2	90 0 255 0	127 911 179 341 539 361 668 445 598 1004
20-10.3	90 0 255 0	127 911 179 341 614 158 790 440 604 995
21-11.1	56 240 210 0	127 911 435 725 873 158 327 490 626 697 860
21-11.2	128 0 410 0	127 911 179 341 614 158 968 283 355 570 625
21-11.3	130 0 396 0	127 911 179 341 539 361 668 445 709 565 1002
22-12.1	77 352 330 0	127 911 435 725 873 158 327 490 626 697 860 932
22-12.2	183 0 600 0	127 911 179 341 539 361 668 173 1002 203 823 589
22-12.3	184 0 594 0	127 911 179 341 614 158 968 283 805 508 604 535
23-13.1	251 0 899 0	127 911 179 341 614 158 968 283 805 466 555 508 535
23-13.2	252 0 890 0	127 911 179 341 614 158 790 440 964 625 995 234 334
23-13.3	252 0 892 0	127 911 179 341 614 158 790 440 364 589 1002 355 692
24-14.1	336 0 1335 0	127 911 179 341 614 158 790 440 964 625 995 234 334 589

Table 13: A Complete Catalog of 2048-Run Designs of Resolution ≥ 7

Design	$(A_7, A_8, \ldots)$	Columns
12-1.1	0 0 0 0 0 1	2047
12-1.2	0 0 0 0 1 0	1023
12-1.3	0 0 0 1 0 0	511
12-1.4	0 0 1 0 0 0	255
12-1.5	0 1 0 0 0 0	127
12-1.6	1 0 0 0 0 0	63
13-2.1	0 1 2 0 0 0	255 1807
13-2.2	0 2 0 1 0 0	511 1567
13-2.3	0 3 0 0 0 0	127 911
13-2.4	1 0 1 1 0 0	511 1551
13-2.5	1 1 0 0 1 0	1023 1055
13-2.6	1 1 1 0 0 0	255 783
13-2.7	2 0 0 0 0 1	2047 63
13-2.8	2 0 0 1 0 0	511 543
13-2.9	2 1 0 0 0 0	127 399
14-3.1	0 7 0 0 0 0	127 911 1459
14-3.2	2 3 2 0 0 0	255 1807 819
14-3.3	3 2 1 1 0 0	511 1567 615
14-3.4	3 3 0 0 1 0	127 911 1175
14-3.5	4 1 0 2 0 0	511 543 1127
14-3.6	4 2 0 0 0 1	2047 63 455
14-3.7	4 3 0 0 0 0	127 911 435
15-4.1	0 15 0 0 0 0	127 911 1459 1749
15-4.2	4 7 4 0 0 0	127 911 1459 725
15-4.3	6 5 2 2 0 0	255 1807 819 1397
15-4.4	6 7 0 0 2 0	127 911 1459 469
15-4.5	7 6 0 0 1 1	127 911 1175 441
15-4.6	7 7 0 0 0 0	127 911 1459 444
15-4.7	8 7 0 0 0 0	127 911 435 725
16-5.1	0 30 0 0 0 0	127 911 1459 1749 1897
16-5.2	7 15 8 0 0 0	127 911 1459 1749 444
16-5.3	8 14 8 0 0 0	127 911 1459 725 873
16-5.4	10 15 0 0 6 0	127 911 1459 1749 470
16-5.5	11 11 4 4 0 0	127 911 1459 725 444
16-5.6	12 13 0 0 4 2	127 911 1459 469 694
16-5.7	15 15 0 0 0 0	127 911 435 725 873
17-6.1	16 30 0 0 16	127 911 1459 1749 1897 470
17-6.2	20 25 0 0 12	127 911 1459 1749 470 739
17-6.3	21 25 0 0 10	127 911 1459 469 694 876
18-7.1	32 46 0 0 32	127 911 1459 1749 1897 470 739
18-7.2	33 45 0 0 30	127 911 1459 1749 470 739 826
19-8.1	52 78 0 0 72	127 911 1459 1749 1897 470 739 826
20-9.1	80 130 0 0	127 911 1459 1749 1897 470 739 826 1272
21-10.1	120 210 0 0	127 911 1459 1749 1897 470 739 826 1272 1309
22-11.1	176 330 0 0	127 911 1459 1749 1897 470 739 826 1272 1309 1614
23-12.1	253 506 0 0	127 911 1459 1749 1897 470 739 826 1272 1309 1614 1956

Table 14: A Complete Catalog of 4096-Run Designs of Resolution ≥ 8

Design	$(A_8, A_9, \ldots)$	Columns
13-1.1	0 0 0 0 0 1	4095
13-1.2	0 0 0 0 1 0	2047
13-1.3	0 0 0 1 0 0	1023
13-1.4	0 0 1 0 0 0	511
13-1.5	0 1 0 0 0 0	255
13-1.6	1 0 0 0 0 0	127
14-2.1	0 2 1 0 0 0	511 3615
14-2.2	1 0 2 0 0 0	511 3599
14-2.3	1 1 0 1 0 0	1023 3103
14-2.4	1 2 0 0 0 0	255 1807
14-2.5	2 0 0 0 1 0	2047 2111
14-2.6	2 0 1 0 0 0	511 1567
14-2.7	3 0 0 0 0 0	127 911
15-3.1	3 4 0 0 0 0	255 1807 2867
15-3.2	5 0 2 0 0 0	511 3599 1651
15-3.3	6 0 0 0 1 0	2047 2111 2503
15-3.4	7 0 0 0 0 0	127 911 1459
16-4.1	7 8 0 0 0 0	255 1807 2867 3413
16-4.2	11 0 4 0 0 0	511 3599 1651 2741
16-4.3	13 0 0 0 2 0	2047 2111 2503 2777
16-4.4	14 0 0 0 0 0	127 911 1459 2492
16-4.5	15 0 0 0 0 0	127 911 1459 1749
17-5.1	14 16 0 0 0 0	255 1807 2867 3413 3734
17-5.2	15 15 0 0 0 0	255 1807 2867 3413 3689
17-5.3	22 0 8 0 0 0	511 3599 1651 2741 3289
17-5.4	25 0 0 0 6 0	2047 2111 2503 2777 2922
17-5.5	30 0 0 0 0 0	127 911 1459 1749 1897
18-6.1	45 0 0 0 18 0	2047 2111 2503 2777 2922 3308
18-6.2	46 0 0 0 16 0	2047 2111 2503 2777 2922 2996
19-7.1	78 0 0 0 48 0	2047 2111 2503 2777 2922 3308 2996
20-8.1	130 0 0 0 120	2047 2111 2503 2777 2922 3308 2996 3441
21-9.1	210 0 0 0 280	2047 2111 2503 2777 2922 3308 2996 3441 3482
22-10.1	330 0 0 0 616	2047 2111 2503 2777 2922 3308 2996 3441 3482 3670
23-11.1	506 0 0 0 1288	2047 2111 2503 2777 2922 3308 2996 3441 3482 3670 3747
24-12.1	759 0 0 0 2576	2047 2111 2503 2777 2922 3308 2996 3441 3482 3670 3747 3853