The Structure of Optimal Design Algorithms

Randall D. Tobias
Senior Research Statistician
SAS Institute, Inc.
Box 8000, Cary, North Carolina, 27511.

1 Introduction

1.1 Combinatorial Design *versus* "Optimal" Design

The practise of theoretical design has not traditionally been one of those areas of applied mathematics where well-posed questions lead directly to well-defined answers. Instead, combinatorial arrangements are typically first proposed as designs on more or less obscure and intuitive grounds; various precisely defined notions of design optimality might be employed, but almost exclusively to *confirm* that a design is good, rather than to *generate* the design in the first place. But fortuitous combinatorial arrangements are not always available: when they are not, classical design theory leaves the practitioner out in the cold.

On the other hand, there are several algorithms for searching for designs which are optimal according to a precise efficiency criterion: the DETMAX algorithm of Mitchell (1974a) is standard; Cook and Nachtsheim (1980) compare several. Typically, they are used to select from a **given** set of candidate points a set of points of a **given** size which optimize a **given** criterion—for example, the determinant of the information matrix $X'X$ with respect to a **given** linear model. While such algorithms are often implemented in packages for experimental design, they don't seem to be in common use, and all those **given**'s in the last sentence indicate why this might be the case. *Algorithmically optimal* designs are good only for precisely defined situations: there is no guarantee they will be good even for other optimality criteria. (In fact, while search algorithms invariably find efficient designs, they may fail to find *the* precisely optimal one even for the given criterion.) By contrast, classical designs, with their high degree of combinatorial structure, tend to be efficient for a variety of reasons.

Thus, algorithmic optimality has by no means made classical combinatorial design theory obsolete, though it does provide answers in many situations for which there are no standard designs. Part of the purpose of this paper is to point out that optimal design search programs can be used in searching for combinatorial designs—that is, they can be used as *generators* of arrangements which are likely to have a high degree of combinatorial structure. This point of view results in an *empirical* approach to theoretical design, based on searching for structure in the results of an optimal design search. The term *experimental mathematics* has been coined to refer to similarly computer-dependent studies in fractal geometry; and indeed the search for structure in combinatorial arrangements is much like the search for rules of non-linear dynamics which account for the beauty of the Mandelbrot and related sets.

1.2 Small Second-order Designs

Use of an optimal search program to explore the general structure of a class of designs is reported in Mitchell (1974b), where small first-order two-level designs are studied. Mitchell used the DETMAX algorithm to generate optimal first-order designs for up to 9 factors, and found that the best design can almost always be constructed from a near-by orthogonal design by adding or deleting 1 or 2 points. The Plackett-Burman/Hadamard matrix designs provide orthogonal first order designs for all practical design sizes $N = 4t$.

Turning to the problem of small second-order designs, the first thing to notice is that orthogonal designs are in this case few and far between. In fact, only the orthogonal 2_V^{k-p} series of regular fractions is known: this provides, for example, a design in 64 runs for 7 factors, while there are only 29 parameters in the model. Rechtschaffner (1967) gave a general method for obtaining a saturated fraction for any number of factors k. The designs in this series are constructed in terms of the following subsets of all 2^k possible combinations of k $+$'s and $-$'s:

$$2_i^k \quad = \quad \{x \in 2^k : x \text{ contains } i \text{ +'s}\}.$$

Then Rechtschaffner's designs can be concisely written as

$$2_1^k \cup 2_{k-2}^k \cup 2_k^k. \tag{1}$$

Srivastava and Chopra (1971) showed that for $4 \leq k \leq 6$, designs of the form (1) are *balanced* in the sense that the variance matrix of the estimates is invariant under any permutation of the factors; and that within the class of balanced designs they are in fact A-optimal. However, it's not known how good these designs are in general, ie. for $k > 6$. Besides this, there has been a fair amount of work on small designs for *response surface* experiments (where quadratic terms as well as linear terms are included), but little of it is directly applicable to the construction of two-level designs in particular. For example, Draper and Lin (1990) have proposed selecting columns of a Plackett-Burman design for use as the factorial portion of a central composite design, but they do not study how well these designs fare by themselves as second-order two-level designs.

This paper concentrates on finding *saturated* second-order two-level designs, where the number of runs is equal to the number of parameters to be estimated ($1+k(k+1)/2$ for k factors). While saturated designs leave no degrees of freedom for error when all terms of the second-order model are retained, in practice only a few of the interaction terms are usually needed. By applying an optimal design search procedure to the problem and exploring the structure of the results, a series of saturated second-order two-level designs is discovered, related to (1) but uniformly better for $k > 6$. The designs in this series are fully efficient up to 7 factors, and moderately efficient up to at least 12 factors.

Sections 2 and 3 exhibit the use of computational facilities to generate optimal designs and explore their structure. In section 4 we use these techniques to discover the new class of saturated second-order designs. The OPTEX procedure of SAS/QC® software, Version 6, is used to search for optimal designs; various other SAS facilities are also used both for generating and exploring these designs.

2 The OPTEX Procedure

It is quite easy to generate optimal saturated second-order designs using SAS/QC® software. The two relevant procedures are FACTEX and OPTEX. FACTEX is designed to construct regular q^k fractions from input model specifications, but it can also be used to create the full 2^k candidate set; then the OPTEX procedure is used to select an optimal subset. For example, the following code uses the two procedures to find a D-optimal saturated second-order design for 5 factors.

```
PROC FACTEX;
   FACTORS A B C D E;             Declare the factors.
   OUTPUT OUT=FULL;               Output the 2^5 to the dataset FULL.
PROC OPTEX;
   MODEL A|B|C|D|E@2;             Declare a second-order model.
   GENERATE N=SATURATED;          Ask for a saturated design.
RUN;
```

The printed results are a listing of relative efficiencies of the designs found in several tries:

```
                                                              Prediction
Design                                                         Standard
Number     D-efficiency     A-efficiency     G-efficiency        Error
------------------------------------------------------------------------
     1       100.0000         100.0000         100.0000          1.0000
     2       100.0000         100.0000         100.0000          1.0000
     3       100.0000         100.0000         100.0000          1.0000
     4       100.0000         100.0000         100.0000          1.0000
     5       100.0000         100.0000         100.0000          1.0000
     6       100.0000         100.0000         100.0000          1.0000
     7       100.0000         100.0000         100.0000          1.0000
     8        71.7594          46.5183          43.7595          1.4662
     9        71.7594          46.5183          43.7595          1.4662
    10        71.7594          45.7304          42.9547          1.4788
```

Because of a certain amount of randomness in the starting design for the search, the procedure makes 10 tries by default. In 7 of these 10, the orthogonal 2_V^{5-1} design was found. (See the appendix for the definitions of the efficiency factors.)

3 Exploring Design Structure

The smallest non-trivial saturated second-order design is for 4 factors in 11 runs. (For 2 factors the full 2^2 is required to estimate the parameters. For 3 factors, there are 7 parameters in the model, so that 7 of the 8 different 2^3 points are required: clearly, any 7 will do.) In searching for combinatorial structure, an obvious first step is to look for some pattern in the actual combinations of +'s and −'s of the optimal design. Converting the above program to 4 factors, the following design results:

A	B	C	D
−	−	−	−
−	−	−	+
−	−	+	−
−	+	−	+
−	+	+	−
−	+	+	+
+	−	−	+
+	−	+	−
+	−	+	+
+	+	−	−
+	+	+	+

No patterns are immediately apparent. However, the efficiency of the design is invariant to changes in the *signs* of the factors, and this may well mask any pattern. Another option is to examine the $X'X$ matrix itself, since it is upon this that the efficiency of the design depends. (The SAS/IML® matrix programming language was used to examine $X'X$ under a variety of permutations of the factors and the factor levels.)

		A	B	C	D	AB	AC	AD	BC	BD	CD
	11	-1	-1	1	1	-1	1	1	1	1	-1
A	-1	11	-1	1	1	-1	1	1	-3	-3	3
B	-1	-1	11	1	1	-1	-3	-3	1	1	3
C	1	1	1	11	-1	-3	-1	3	-1	3	1
D	1	1	1	-1	11	-3	3	-1	3	-1	1
AB	-1	-1	-1	-3	-3	11	1	1	1	1	3
AC	1	1	-3	-1	3	1	11	-1	-1	3	1
AD	1	1	-3	3	-1	1	-1	11	3	-1	1
BC	1	-3	1	-1	3	1	-1	3	11	-1	1
BD	1	-3	1	3	-1	1	3	-1	-1	11	1
CD	-1	3	3	1	1	3	1	1	1	1	11

The first row and column correspond to the intercept. If factors C and D are multiplied by -1, the information matrix for the main effects will be of the form $aI + bJ$, where I is the identity and J the all-ones matrix—an auspicious combination! Once this is done, a pattern does emerge in the design.

A	B	$-C$	$-D$
$+$	$-$	$-$	$-$
$-$	$+$	$-$	$-$
$-$	$-$	$+$	$-$
$-$	$-$	$-$	$+$
$-$	$-$	$+$	$+$
$-$	$+$	$-$	$+$
$-$	$+$	$+$	$-$
$+$	$-$	$-$	$+$
$+$	$-$	$+$	$-$
$+$	$+$	$-$	$-$
$+$	$+$	$+$	$+$

The design is composed of all combinations with one or two $+$'s, plus the combination of all four $+$'s. Symbolically,

$$O_4 = 2^4_1 \cup 2^4_2 \cup 2^4_4. \tag{2}$$

For 5 factors the optimal design is known, the 2_V^{5-1} fraction. Can it also be articulated in terms of the subsets 2_i^5? The following program uses FACTEX in its intended capacity to generate the design, and then prints it out sorted by the number of +'s in each run.

```
PROC FACTEX;
   FACTORS A B C D E;
   MODEL RESOLUTION=5;                 Declare the desired resolution
   SIZE DESIGN=16;                       and the design size.
   OUTPUT OUT=D5;
DATA D5; SET D5;
   NPLUS = (SUM(A B C D E) + 5)/2;     Compute the number of +'s.
PROC SORT;  BY NPLUS A B C D E;
PROC PRINT; BY NPLUS;
RUN;
```

The 2_V^{5-1} design does indeed have a structure similar to O_4—namely,

$$O_5 \quad = \quad 2_V^{5-1} \quad = \quad 2_1^5 \cup 2_3^5 \cup 2_5^5. \tag{3}$$

Finally, for 6 factors the OPTEX procedure is used again, and examination of the $X'X$ matrix is again required in order to scale the factor levels appropriately. When this is done, the optimal saturated second-order design for 6 factors has the form

$$O_6 \quad = \quad 2_1^6 \cup 2_4^6 \cup 2_6^6. \tag{4}$$

Thus, techniques which can be used to explore the combinatorial structure of a given arrangement include examining the actual points as well as the information matrix, possibly under permutations of the factors and/or their respective levels. It's also handy to be able easily to compute and organize the design by any functions which may appear important (in this case, the number of +'s). Using these techniques, we have "found" the balanced combinatorial structure of Rechtschaffner's series of designs (1) for k=4, 5, and 6.

4 A Series of Efficient Saturated Second-order Designs

For $k = 7$ factors, however, an optimal design search finds a better design than (1) yields. After appropriate permutations of the factor levels, the part of the $X'X$ matrix of this design corresponding to main effects displays a *partially balanced* structure, shown below:

	A	B	C	D	E	F	G
	29	5	5	1	1	1	1	1
A	5	29	5	1	1	1	1	1
B	5	5	29	1	1	1	1	1
C	1	1	1	29	5	5	5	5
D	1	1	1	5	29	5	5	5
E	1	1	1	5	5	29	5	5
F	1	1	1	5	5	5	29	5
G	1	1	1	5	5	5	5	29

It isn't possible to make this invariant to permutations of the factors, so the design isn't balanced in the sense of Srivastava and Chopra (1971) and in particular it can't be written as a union of 2^7_i terms. It does appear that the $k+1$ runs $2^k_1 \cup 2^k_k$ are still in the design; but instead of 2^k_{k-2}, the "left-over" part now takes the form

$$O_7 - \left(2^7_1 \cup 2^7_7\right) \quad = \quad \left\{2^7_5 \; - \; (++) \oplus 2^5_3\right\} \; \cup \; \left\{(++) \oplus 2^5_2\right\}, \tag{5}$$

where $(++) \oplus 2^5_3$, for example, means the runs for which the first two factors are both $+$ and the rest are chosen from 2^5_3; $\{2^7_5 \; - \; (++) \oplus 2^5_3\}$ means all runs in 2^7_5 except those for which the first two factors are both $+$.

Note that by exchanging $+$'s and $-$'s we have $2^5_2 = -2^5_3$. But 2^5_3 was the left-over part for the 5 factor design. This observation inspires a guess at the general structure.

$$D_k \quad = \quad 2^k_1 \cup 2^k_k \cup A_k \tag{6}$$

where the left-over part A_k is *recursively* defined as 2^k_2 for $k = 2, 3$, and for $k > 3$,

$$A_k \quad = \quad \left\{2^k_{k-2} \; - \; (++) \oplus 2^{k-2}_{k-4}\right\} \; \cup \; \left\{(++) \oplus -A_{k-2}\right\}. \tag{7}$$

From (5) it's clear that O_7 is of the form (6). Furthermore, if $A_{k-2} = 2^{k-2}_2 = -2^{k-2}_{k-4}$ then $A_k = 2^k_{k-2}$, so that (6) matches (1) for $4 \le k \le 6$ also. It also gives designs of the right size, as the next result shows.

Theorem 1 *For all $k \ge 3$, $|D_k| \; = \; 1 \; + \; k(k+1)/2$.*

Proof:

First note that

$$\begin{aligned} |2^k_1| &= k \\ |2^k_k| &= 1 \\ |2^k_2| &= k(k-1)/2 \qquad (8) \\ &= |2^k_{k-2}|. \qquad (9) \end{aligned}$$

Table 1: Efficiencies of D_k Relative to DETMAX

Number of Factors	Number of Runs	Efficiency Factors		
		D	A	G
4	11	100%	100%	100%
5	16	100%	100%	100%
6	22	100%	100%	100%
7	29	102%	110%	126%
8	37	94%	109%	149%
9	46	89%	104%	129%
10	56	78%	97%	155%
11	67	70%	88%	136%
12	79	63%	73%	116%

Then $|D_k| = 1 + k + |A_k|$ and the theorem will be proved if we can show that

$$|A_k| \;=\; k(k-1)/2. \tag{10}$$

Now, by (8), (10) clearly holds for $2 \leq k \leq 6$, where $A_k = 2^k_2$ or $A_k = 2^k_{k-2}$. Then

$$\begin{array}{rclllll} |A_k| & = & |2^k_{k-2}| & - & |2^{k-2}_{k-4}| & + & |A_{k-2}| & \\ & = & \frac{k(k-1)}{2} & - & \frac{(k-2)(k-3)}{2} & + & |A_{k-2}| & \text{by (9)}, \\ & = & \frac{k(k-1)}{2} & - & \frac{(k-2)(k-3)}{2} & + & \frac{(k-2)(k-3)}{2} & \text{by induction}, \\ & = & \frac{k(k-1)}{2}. & & & & & \end{array}$$

From the theorem, $|D_3| = 7$, so that by our remarks at the beginning of Section 3 D_3 is also optimal. Thus, the series of designs D_k recursively defined by (6) and (7) gives optimal saturated second-order designs for all $3 \leq k \leq 7$. How about for $k > 7$? Tables 1 and 2 show the efficiency of D_k relative to O_k and (1), respectively, for $4 \leq k \leq 12$. (Note: The efficiencies in Table 1 for the 7 factor design are greater than 100% because the table was generated using the standard 10 tries of the DETMAX algorithm, whereas it required 100 tries of a much slower algorithm, the "Modified Federov" algorithm of Cook and Nachtsheim (1980), to find D_7 using OPTEX.) Table 1 shows that the series of designs D_k is not absolutely optimal for $k > 7$, but it also shows that an absolutely optimal design is not available for these situations, since D_k has worse D-efficiency but better G-efficiency than the D-optimal design. On the other hand, Table 2 shows that D_k is uniformly better than the corresponding design generated according to (1) for $k > 7$.

Table 2: Efficiencies of D_k Relative to Rechtschaffner's Designs

Number of Factors	Number of Runs	Efficiency Factors		
		D	A	G
4	11	100%	100%	100%
5	16	100%	100%	100%
6	22	100%	100%	100%
7	29	108%	111%	104%
8	37	112%	115%	102%
9	46	120%	124%	105%
10	56	125%	127%	103%
11	67	132%	133%	105%
12	79	136%	135%	103%

The design D_7 is not balanced in the sense of Srivastava and Chopra (1971): the $X'X$ matrix is not invariant to arbitrary permutations of the factors. However, it is *partially* balanced in a certain sense: the factors can be partitioned so that the $X'X$ matrix is invariant to permutations within each partition. Now, the balanced designs of Srivastava and Chopra (1971) are of the form

$$\bigcup_i 2_i^k.$$

The above observation suggests broadening the exploration for optimal second-order designs to designs of the form

$$\bigcup_{i,j} 2_i^m \oplus 2_j^n,$$

where $m + n = k$.

5 Conclusion

We have shown how computer experimental design facilities can be used to search for designs of the more traditional combinatorial type. By way of example, we have used standard experimental design and data manipulation tools to discover construction rules for a series of efficient saturated second-order factorial designs. The designs in this series exhibit a certain degree of partial balance which may be a fruitful combinatorial characteristic to pursue in further research.

6 References

Cook, R.D. and Nachtsheim C.J. (1980). "A Comparison of Algorithms for Constructing Exact D-optimal Designs". *Technometrics* 22, pp. 315-324.

Draper, N.R., and Lin, D.K.J. (1990). "Small Response Surface Designs," *Technometrics*, 32:2, pp. 187-194.

Dykstra, O.Jr. (1971). "The Augmentation of Experimental Data to Maximize $|X'X|$". *Technometrics* 13, pp. 682-688.

Federov, V.V. (1972). *Theory of Optimal Experiments,* translated and edited by W.J. Studden and E.M. Klimko. New York: Academic Press.

Galil, Z. and Kiefer, J. (1980). "Time- and Space-saving Computer Methods, Related to Mitchell's DETMAX, for Finding D-Optimum Designs." *Technometrics* 22, pp. 301-313.

Mitchell, T.J. (1974a). "An Algorithm for the Construction of D-optimal Experimental Designs". *Technometrics* 20, pp. 203-210.

Mitchell, T.J. (1974b). "Computer Construction of 'D-optimal' First-Order Designs". *Technometrics* 20, pp. 211-220.

Rechtschaffner, R.L. (1967). "Saturated Fractions of 2^n and 3^n Factorial Designs". *Technometrics* 9, pp. 569-575;

Srivastava, J.N. and Chopra, D.V. (1971). "Balanced Optimal 2^m Fractional Factorial Designs of Resolution V, $m \leq 6$." *Technometrics* 13, pp. 257-269.

A Definition of Efficiency Factors

D	$100 \times \frac{\|X'X\|^{1/p}}{N_D}$
A	$100 \times \frac{p}{\text{trace}(N_D \times (X'X)^{-1})}$
G	$100 \times \frac{\sqrt{p/N_D}}{\sigma_M}$

where p is the number of independent variables in the linear model, N_D is the number of points in the target design, and σ_M is the maximum standard error for prediction over the candidate points. For the saturated second-order designs which we consider,

$$p \; = \; N_D \; = \; 1 + k + k(k-1)/2$$

where k is the number of factors. The D- and A-efficiencies are the relative number of runs (expressed as percents) required by a hypothetical orthogonal design to achieve the same $|X'X|$ and trace $((X'X)^{-1})$, respectively. (See Mitchell (1974b).) The numerator in the G-efficiency is a lower bound on σ_M for the optimal design.