A Comparison of Algorithms for Constructing Exact D-Optimal Designs
Author(s): R. Dennis Cook and Christopher J. Nachtsheim
Source: *Technometrics*, Vol. 22, No. 3 (Aug., 1980), pp. 315-324
Published by: Taylor & Francis, Ltd. on behalf of American Statistical Association and American Society for Quality
Stable URL: http://www.jstor.org/stable/1268315
Accessed: 27-06-2016 04:14 UTC

REFERENCES

Linked references are available on JSTOR for this article:
http://www.jstor.org/stable/1268315?seq=1&cid=pdf-reference#references_tab_contents
You may need to log in to JSTOR to access the linked references.

TECHNOMETRICS©, VOL. 22, NO. 3, AUGUST 1980

A Comparison of Algorithms for Constructing Exact D-Optimal Designs

R. Dennis Cook
Department of Applied Statistics
University of Minnesota
St. Paul, MN 55108

Christopher J. Nachtsheim
Statistics Group
Los Alamos Scientific Laboratory
Los Alamos, NM 87545

An empirical comparison of existing algorithms for the computer generation of exact D-optimal experimental designs is carried out. Among algorithms considered were those due to Wynn, Mitchell, Fedorov, and Van Schalkwyk. A procedure for rounding off approximate designs as suggested by Kiefer is also evaluated. A modification of the Fedorov algorithm is given and shown to effect substantial decreases in the computer time required for design generation.

KEY WORDS

Computer search
D-optimality
Exchange algorithm
Mixture designs
Optimal designs

1. INTRODUCTION AND NOTATION

Recent interest in the use of exact D-optimal experimental designs has spawned a variety of iterative methods for their construction. Many of the early approaches involved the direct maximization of $|\mathbf{X}'\mathbf{X}|$ by mathematical programming techniques (Box, 1966, Atkinson, 1969, Hartley and Ruud, 1969, Mahoney, 1970, Box and Draper, 1971, Newhardt and Bradley, 1971). Because of problems inherent in this type of approach (e.g., dimensionality) attention has more recently focused on a number of methods which take into account some of the special characteristics of the design problem.

The ability to compute approximate D-optimal designs iteratively (Fedorov, 1969) led Kiefer (1971) to suggest simply rounding off the approximate D-optimal design to obtain the exact design. Kiefer noted that for large N, N-point designs obtained in this way are very nearly D-optimal.

The bulk of the remaining approaches may be classified as exchange algorithms. These procedures begin with nonsingular N-point designs and then add and delete one or more observations in order to achieve increases in the determinant. Three such algorithms are well known: The first of these was developed by Fedorov (1969, 1972). The Fedorov algorithm discussed here is distinct from Fedorov's well-known algorithm for generating approximate D-optimal designs. The second (henceforth referred to as the Wynn-Mitchell algorithm) is due independently to Mitchell and Miller (1970) and to Wynn (1972). Both the Fedorov and Wynn-Mitchell algorithms are single point exchange algorithms. In contrast, Mitchell's (1974) algorithm allows more than a single point to be added and deleted at each iteration. Other algorithms have been proposed (see, e.g., Van Schalkwyk, 1971) which reflect variations in the basic Fedorov algorithm.

Despite widespread use, little is generally known and nothing has been written on the relative performance of the existing procedures for finding discrete designs. In both St. John (1973, p. 156) and St. John and Draper (1975, p. 20), the need is indicated for a detailed comparison of the basic exchange procedures. This note summarizes research undertaken to determine the relative performance characteristics of the following:

1. The Kiefer round-off procedure.
2. The Fedorov algorithm.
3. The Wynn-Mitchell algorithm.
4. The Van Schalkwyk algorithm.
5. The Mitchell algorithm.
6. A modified Fedorov algorithm.
7. A combined Fedorov, Wynn-Mitchell algorithm.

Received September 1979; revised April 1980

The bases for procedures 6 and 7 grew from early computational experience with the Fedorov and Wynn-Mitchell algorithms. Each of the seven algorithms is detailed in Section 2. In all phases of this study, attempts have been made to adhere to the guidelines set forth by Crowder, Dembo, and Mulvey (1979) on the reporting of computational experiments with mathematical software.

Throughout, we assume that an observation $y_i(x_i)$ may be written

$$y_i = \mathbf{f}'(x_i)\boldsymbol{\theta} + e_i \qquad i = 1, 2, \cdots, N, \tag{1}$$

where the x_i's are elements of a compact design space, χ, $\mathbf{f}$ is continuous on χ, and the e_i's are uncorrelated random variables with mean zero and variance σ^2. Letting $\mathbf{f}'(x_i)$ denote the $(m \times 1)$ i^{th} row of a matrix $\mathbf{X}$, (1) may be rewritten

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \mathbf{e}.$$

A design ξ_N is an N-point exact design if ξ_N is a probability measure on χ and $N\xi_N(x)$ is a nonnegative integer for every $x \in \chi$. Denote the space of N-point exact designs on χ by Ξ_χ^N. For approximate designs, the restriction that $\xi(x)$ be a multiple of $1/N$ is dropped. Thus, an approximate design is simply an element of the space, Ξ_χ, of probability measures on χ. If $\xi(x) > 0$, x is said to be an element of the support or spectrum, $S(\xi)$, of the design ξ.

Exact and approximate D-optimal experimental designs maximize, for $\xi \in \Xi_\chi^N$ and $\xi \in \Xi_\chi$ respectively, the determinant of the information matrix,

$$\mathbf{M}(\xi) = \int_\chi \mathbf{f}(x)\mathbf{f}'(x)d\xi(x).$$

If $|\mathbf{M}(\xi)| \neq 0$, the dispersion matrix, $\mathbf{M}^{-1}(\xi)$, is written $\mathbf{D}(\xi)$. Note that for an N-point exact design ξ_N,

$$\mathbf{M}(\xi_N) = \mathbf{X}'\mathbf{X}/N.$$

Hence, if we set $N\xi_N = \xi[N]$, we have

$$\mathbf{M}(\xi[N]) = \mathbf{X}'\mathbf{X}.$$

The variance of the least squares predictor $\hat{y}(x)$ is given by

$$\begin{aligned}\operatorname{Var}(\hat{y}(x)) &= \sigma^2\mathbf{f}'(x)\mathbf{D}(\xi[N])\mathbf{f}(x) \\ &= \sigma^2 d(x, \xi[N]),\end{aligned}$$

where $d(x,\xi)$ is defined to be the variance function of the design ξ. The covariance function is defined similarly:

$$d(x_i, x_j, \xi) = \mathbf{f}'(x_i)\mathbf{D}(\xi)\mathbf{f}(x_j).$$

2. EXPLANATION OF METHODS

2.1 The Kiefer Round-off Procedure

Let ξ_D denote an approximate D-optimal design for a model $\mathbf{f}$ on a design space χ with finite support $S = \{x_1, x_2, \cdots, x_s\}$. Using a result given by Keifer (1971, p. 116), if ξ_N^* is chosen in such a way that

$$\inf_{\xi_N \in \Xi_\chi^N} \max_{x \in S} |\xi_N(x) - \xi_D(x)| = \max_{x \in S} |\xi_N^*(x) - \xi_D(x)|, \tag{2}$$

then

$$\frac{\ln|\mathbf{M}(\xi_N^*)|}{\ln|\mathbf{M}(\xi_D)|} = 1 + 0(N^{-2}).$$

We note that our implementation of the procedure implied by (2) does not explicitly consider the maximization of $|\mathbf{M}(\xi_N)|$. In fact, many designs could satisfy (2), with some being better than others in terms of the determinant criterion. No attempt is made to find the best such design.

2.2 The Basic Exchange Procedure: Fedorov's Algorithm

The Fedorov algorithm starts with an N-point nonsingular design, $\xi_1[N]$. During the i^{th} iteration a point, say x_j, is deleted and another point $x \in \chi$ is added in such a way that the resulting increase in the determinant is maximal. Fedorov showed that

$$\frac{|\mathbf{M}(\xi_{i+1}[N])|}{|\mathbf{M}(\xi_i[N])|} = 1 + \Delta_i(x_j, x)$$

where

$$\begin{aligned}\Delta_i(x_j, x) &= [-d(x_j, \xi_i[N])] + [d(x, \xi_i[N])] \\ &\quad - d(x, \xi_i[N])d(x_j, \xi_i[N]) + d^2(x_j, x, \xi_i[N])],\end{aligned} \tag{3}$$

herein referred to as the "delta function." We note that if $\xi_i^x[N+1]$ is used to denote the design $\xi_i[N]$ augmented by the point x, then using Lemma 4.2.1 from Federov (1972, p. 175), (3) may be rewritten

$$\begin{aligned}\Delta_i(x_j, x) &= [1 + d(x, \xi_i[N])] \\ &\quad \times [d(x, \xi_i^x[N+1]) - d(x_j, \xi_i^x[N+1])].\end{aligned} \tag{4}$$

In order to achieve steepest ascent, the points x_j and x^* satisfying

$$\max_{x_j \in S(\xi_i[N])} \max_{x \in \chi} \Delta_i(x_j, x) = \Delta_i(x_j, x^*) \tag{5}$$

must be found at each iteration; hence, a maximization of the delta function must be carried out for each support point. The procedure terminates whenever $(|\mathbf{M}(\xi_{i+1}[N])| - |\mathbf{M}(\xi_i[N])|)/|\mathbf{M}(\xi_i[N])|$ becomes sufficiently small. (Algorithms in the current study terminated whenever the value of the above ratio dropped below 10^{-5}.)

2.3 The Wynn-Mitchell, Van Schalkwyk and Mitchell Algorithms

Each of the algorithms described in this section may be viewed as an approximation of Fedorov's

basic exchange procedure. Differences arise mainly in the handling of the optimization of the delta function. The points x_i and x to be exchanged are no longer chosen simultaneously.

In both the Wynn-Mitchell and Van Schalkwyk algorithms the maximization of the delta function is broken into two parts. In the Wynn-Mitchell algorithm at iteration i, the point $x^* \in \chi$ maximizing the first bracketed term in (4) is chosen initially, giving $\xi_i^{x^*}[N+1]$. Given x^*, x_i is then chosen to maximize the second term. Similarly, in the Van Schalkwyk procedure, x_i is first found to maximize $-d(x_j, \xi_i[N])$, the first term in (3). The point $x^* \in \chi$ is subsequently chosen to maximize the second term. Thus, while neither the Wynn-Mitchell nor the Van Schalkwyk algorithm can be considered steepest ascent, only a single maximization of the delta function is needed for an exchange.

In 1974, Mitchell published a generalized version of his earlier algorithm (Wynn-Mitchell) which he called DETMAX. In DETMAX, the requirement that an $(N+1)$-point design be returned immediately to an N-point design is relaxed. The algorithm makes use of positive and negative "excursions" during which the size of the design may vary from N to $N+K$ and from N to $N-K$, respectively, where K is a user-selected integer constant. Basically, the algorithm at first emulates the Wynn-Mitchell algorithm in allowing the design to vary from size N to $N \pm 1$ (excursion size = 1). As the current best N-point design approaches optimality it is necessary to allow larger and larger excursions in order to effect improvements. When an excursion size reaches K, the algorithm terminates. Mitchell recommended setting K equal to 6 for discrete, finite design spaces. In the present study, all design spaces are convex. For such spaces, a maximum excursion size of $K = 4$ has proven effective.

2.4 *A Modified Fedorov Algorithm*

During the initial testing of the above algorithms, it quickly became evident that, in relation to the Wynn-Mitchell and Van Schalkwyk algorithms, the Fedorov procedure is an expensive one to use. The problem stems from the fact that, at each iteration, N maximizations of the delta function are called for. Since the results of only one of these optimizations is ever used, the algorithm is clearly inefficient with respect to the utilization of the extrema of the delta function.

Fedorov (1972, p. 170) briefly addressed the problem in noting that the maximization of the delta function could be "carried out not to the end, but stopped as soon as $\Delta(x_i, x)$ exceeds some given level," say δ. He allowed that this modification would increase the number of iterations, but reduce the average volume of computation. There are, however, problems with this approach. Clearly, an effective value for δ must be determined. If δ is too large, little will be gained in the way of reduced computation. When δ is small, termination of the search procedure before convergence to a local maximum will often be possible, expecially during the early stages of the algorithm. Such points cause immediate increases in the determinant of the information matrix, but may well have to be removed during later stages. In addition, no matter what value of $\delta > 0$ is chosen, the results of many time consuming optimizations of the delta function will be wasted.

A new algorithm was developed to make use of information from the optimization of the delta function normally discarded by the Fedorov algorithm. Each iteration corresponds to an iteration of the Fedorov algorithm. However, an iteration s is broken down into N stages, one for each support point in the design at the start of the iteration. At stage i, the first argument of the delta function is fixed at the ith support point, the point $x^* \in \chi$ is found such that

$$\max_{x \in X} \Delta_s(x_i, x) = \Delta_s(x_i, x^*),$$

and then x_i is exchanged for x^* (i.e., the design is updated). The support points may be randomly ordered at the start of an iteration. A single iteration consists of a number of exchanges equal to the number of support points at the start of the iteration. Since

$$\max_{x \in X} \Delta_s(x_i, x) \geq \Delta_s(x_i, x_i) = 0,$$

an exchange will never result in a decrease in the determinant of the information matrix.

The suggested modification avoids a number of the problems inherent in Fedorov's suggestion. Firstly, none of the optimizations of the delta function are "wasted." If for any j

$$\max_{x \in X} \Delta_s(x_j, x) = \Delta_s(x_j, x^*) > 0,$$

then an improvement is to be made by the exchange of x_j and x^*. The only cost incurred in making the exchange is the extra central processing time necessary to update the design, the dispersion matrix, and the determinant. Experience with the algorithm has indicated that the time necessary to exchange the points x_j and x^* is insignificant in comparison with the time required for a typical maximization of the delta function. Secondly, each optimization of the delta function is carried out until termination at a point x^* which is locally optimal. In this way we avoid problems mentioned earlier with the incorporation of points x at which the delta function, $\Delta(x_j, x)$ is not at least locally optimal for some x_j. Points found according to this scheme are less likely to be

exchanged during later iterations. Finally, for implementation of the procedure, it is no longer necessary to specify a δ-value.

2.5 *A Combined Wynn-Mitchell-Fedorov Algorithm*

As mentioned in the preceding section, early experience with the Fedorov and Wynn-Mitchell algorithms indicated that the former was quite expensive, in terms of central processing time, in relation to the latter. Of interest was the fact that Fedorov's procedure required significantly more computing time to reach a design as efficient as the design constructed by the Wynn-Mitchell algorithm. This led to the conjecture that total time for the Fedorov algorithm might be substantially reduced by a prior application of the Wynn-Mitchell routine. In this way, the Wynn-Mitchell and Federov procedures are combined in what we shall herein refer to as the Wynn-Mitchell-Fedorov algorithm.

3. DESIGN OF THE STUDY

In this section, the structure or experimental design of the study is described in detail. Each test case is determined by the values chosen for the following set of variables:

1. Model, $\mathbf{f}(x)$.
2. Design space, χ.
3. Size of the experiment, N.
4. Starting design, $\xi_1[N]$.

In what follows, we enumerate the choices made for each of these variables and briefly motivate their selection.

In order that the set of models chosen reflect a wide range of empirical models, test models were selected from each of the following four classes:

1. Models for one-dimensional polynomial regression.
2. Additive models for response surface experimentation.
3. Non-additive models for response surface experimentation.
4. Models for mixture experiments.

The full set of ten models chosen is given in Table 1.

Three general design spaces are most often mentioned in the literature in connection with the optimal design of experiments. These are the hypercube, the hypersphere and the simplex. In a mixture experiment, the response depends on the relative proportions, x_i, of the components present and not on the total amount of the mixture. The design space for a q-component mixture experiment is the q-1 dimensional simplex given by

$$\chi = \left\{ \mathbf{x}' = (x_1, \cdots, x_q) \mid x_i \geq 0; \quad i = 1, 2, \cdots, q \quad \text{and} \quad \sum_{i=1}^{q} x_i = 1 \right\}.$$

Because the hypersphere is only rarely used in comparison to the hypercube, the design spaces for the models in classes 1-3 were, in all cases, hypercubes defined by

$$\chi = \{\mathbf{x}' = (x_1, \cdots, x_q) \mid -1 \leq x_i \leq 1; \quad i = 1, 2, \cdots, q\}.$$

For each of the exchange algorithms, designs constructed will depend, to a certain extent, on the starting designs used. In order to gauge the effects of using multiple starting designs, a preliminary test was carried out. Each of the Wynn-Mitchell, Mitchell, Fedorov, and modified Fedorov algorithms were executed six times: three for Model 2.1 with $N = 6$ and three for Model 1.2 with $N = 10$. Design points in each of the six starting designs were selected at random from the design space, χ. While differences in terms of time and D-efficiency resulted, for comparative purposes these differences were not large. Furthermore, the rankings in terms of time and D-efficiency were stable. Because of this stability, we concluded that two starting designs for each design problem would suffice for valid comparisons of methods.

A first starting design was chosen from the support of the approximate D-optimal design by selecting points in order of decreasing weight. The motivation for this choice is the same as that which led to the Kiefer round-off procedure: Often exact D-optimal designs are supported on a subset of the approximate D-optimal design. The set of points comprising the second starting design was chosen at random from the design space. If an experimenter has little or no prior knowledge about the D-optimal design, he is likely to use such an initial design. Furthermore, we need to know how each of the algorithms perform when both slight and substantial changes in the starting design are required. For a given model, each algorithm used the same pair of starting designs.

In order to keep the number of test cases from becoming prohibitively large, the number of values of N was restricted to two. For large N, the D-optimal design, ξ_D, is often closely approximated by a design $\xi_N \in \Xi_\chi^N$ and in such a case little is gained with the use of an exchange algorithm. Also, for N a multiple of m, the number of terms in the model, the D-optimal design is also exact D-optimal for half of the considered models (1.1, 1.2, 1.3, 3.2, and 3.3). Thus, the values of N which were of interest were those such that $m < N < 2m$. Those selected for each model have been listed in Table 2.

Transformations due to Atkinson (1969) and Box (1966) enabled the elimination of the constraint sets and subsequent use of Powell's (1964) method of direct search for the maximization of the variance and

TABLE 1—*Test models.*

Type 1.	Polynomial regression on [-1,1].
1.1	3rd degree.
1.2	5th degree.
1.3	8th degree.
Type 2.	Additive Models for Response Surface Experimentation.
2.1	$f'(x) = (1,x_1,x_2,x_3,x_4)$.
2.2	$f'(x) = (1,x_1,x_2,x_1^2,x_2^2)$.
Type 3.	Non-additive Models for Response Surface Experimentation.
3.1	$f'(x) = (1,x_1,x_2,x_1x_2,x_1^2,x_2^2)$.
3.2	$f'(x) = (1,x_1,x_2,x_3,x_1x_2,x_1x_3,x_2x_3,x_1x_2x_3)$
3.3	$f'(x) = (1,x_1,x_2,x_1x_2,x_1^2,x_2^2,x_1x_2^2,x_1^2x_2,x_1^2x_2^2)$
Type 4.	Models for Mixture Experiments.*
4.1	$f'(x) = (x_1,x_2,x_3,x_4,x_1^{-1},x_2^{-1},x_3^{-1},x_4^{-1})$
4.2	$f'(x) = (x_1,x_2,x_3,x_1x_2,x_1x_3,x_2x_3,x_1^{-1},x_2^{-1},x_3^{-1})$

*Draper and St. John (1977).

delta functions. In general, rather than attempting to find global maxima for each call of the Powell algorithm, initial searches are terminated at the first point which is locally optimal. Starting points for the Powell search are selected at random. However, when an exchange algorithm terminates, an extra, more exhaustive search of the design space is made as a second check for termination as follows: A coarse grid is placed over the design space. Using each point in the grid as a starting point, a Powell search is carried out. If incorporation of the best of the resulting set of local maxima results in an improvement in the design, the exchange algorithm is resumed as usual; otherwise, it is allowed to terminate. Note that because of the special characteristics of the termination criterion for Mitchell's algorithm, the combination grid-Powell search was not used in conjunction with DETMAX.

4. RESULTS

Early experience with the Wynn-Mitchell-Fedorov algorithm led to its elimination from the study. Although designed to speed the execution of the Fedorov algorithm by a prior application of the much faster Wynn-Mitchell algorithm, just the opposite occurred. The combined algorithm took consistently more central processing time than did the Fedorov algorithm alone. An explanation for this can be attributed to the fact that the two algorithms did not, in general, find the same designs. Although the starting designs provided by the Wynn-Mitchell algorithm were usually "good" in terms of efficiency, the Fedorov algorithm would change many of the points, if only slightly. As a result, little was gained by the preapplication of the Wynn-Mitchell algorithm.

With the elimination of the combined algorithm, the number of methods in the study was decreased to six. Each of the five exchange algorithms were executed 40 times, corresponding to the 40 test cases constructed from 10 models, 2 starting designs and 2 values of N. The Kiefer procedure uses only the approximate D-optimal design as a starting design and hence is run 20 times. Algorithms were programmed in MNF FORTRAN and executed on a CDC 6400 computer under the KRONOS operating system. A program listing is given in Nachtsheim (1979). For

TABLE 2—*Values of N chosen for each test model.*

Model	m	N(Low)	N(High)
1.1	4	5	7
1.2	6	8	10
1.3	9	12	15
2.1	5	6	9
2.2	5	6	9
3.1	6	8	10
3.2	8	10	14
3.3	9	12	15
4.1	8	10	14
4.2	9	12	15

each of the 220 runs necessary, four variables were recorded: the central processing time (excluding input, output, and preprocessing time), the number of evaluations of the delta or variance functions, and both the determinant and the "relative efficiency" of the resultant design. Relative efficiency, R_i, of the ith design is defined as follows: Let $\xi_i[N]$ be the design constructed by the ith algorithm. Then

$$R_i = \frac{|\mathbf{M}(\xi_i[N])|^{1/m}}{\max_j |\mathbf{M}(\xi_j[N])|^{1/m}} \times 100.$$

Relative efficiency was used instead of D-efficiency because, in all cases, exact D-optimal designs were (and still are) unknown.

In Table 3(a), the average central processing times, average ranks (in terms of *CP* times) and average numbers of function evaluations per execution are listed for each algorithm when the starting design is randomly selected. Table 3(b) presents a similar report for those test cases in which the starting design was chosen from $S(\xi_D)$. For example, for all test cases in which the starting design was chosen at random, the Van Schalkwyk algorithm was fastest, ranking 1.4 on the average, and requiring an average of 15.53 seconds and 1258 function evaluations. Note that the order in which the exchange algorithms are ranked is the same for both groups of test cases. In a similar manner, average relative efficiencies and ranks in terms of relative efficiencies are presented in Table 4. In this case, differences in ranking do exist between the two groups. The positions of the Fedorov and modified Fedorov algorithms are switched, as well as

TABLE 3—*Average execution times, numbers of function evaluations and ranks in terms of execution time.*

(a) Random Starting Designs

Algorithm	Average CP Time	Average Rank (Time)	Average Numbers of Function Evaluations
Van Schalkwyk	15.53	1.40	1258
Wynn-Mitchell	18.31	1.75	1455
Mitchell	46.72	3.05	3432
Modified Fedorov	92.49	3.90	6239
Fedorov*	183.31	4.90	12329

(b) Starting Design from $S(\xi_D)$

Algorithm	Average CP Time	Average Rank (Time)	Average Numbers of Function Evaluations
Kiefer**	.09	1.00	0
Van Schalkwyk	4.37	2.50	322
Wynn-Mitchell	4.52	2.50	317
Mitchell	41.83	4.25	3001
Modified Fedorov	70.60	4.97	4086
Fedorov†	109.77	5.78	5983

*One run terminated prior to convergence after 650 seconds.

**Based on 19 completed runs.

†Two runs terminated prior to convergence after 650 seconds.

TABLE 4—*Average relative efficiencies and ranks in terms of relative efficiencies.*

(a) Random Starting Designs

Algorithm	Average Relative-Efficiency	Average Rank (Relative-Efficiency)*
Modified Fedorov	99.95	1.89
Fedorov	99.92*	2.05
Mitchell	99.42	3.15
Wynn-Mitchell	98.43	3.84
Van Schalkwyk	97.77	4.06

(b) Starting Designs Chosen from $S(\xi_S)$

Algorithm	Average Relative-Efficiency	Average Rank (Relative-Efficiency)**
Fedorov	99.96**	2.76
Modified Fedorov	99.87	3.03
Mitchell	99.47	3.35
Van Schalkwyk	99.41	3.70
Wynn-Mitchell	99.19	3.85
Kiefer	98.82*	4.35

*Based on 19 completed runs.

**Based on 18 completed runs.

those of the Wynn-Mitchell and Van Schalkwyk algorithms. However, since differences in average ranks and relative efficiencies in each of these cases are slight, the mild permutations which result are not surprising. In Table 5, the times and relative efficiencies of each of the exchange algorithms are averaged over all test cases.

Tables similar to 3 and 4, not included here, were prepared for each of the eight model class by starting design combinations. Very few changes resulted in the orderings of the algorithms by average rank, and these occurred only between adjacent algorithms. Because of this, the effect of any interaction between the relative performances of the algorithms and model class (or starting design) is not considered to be significant.

In Tables 6(a) and (b), the execution times of the algorithms are compared on a pairwise basis. The $(i, j)^{th}$ entry in each of the tables gives the number of times that the i^{th} algorithm exceeded the j^{th} algorithm in execution time. Note that the rankings of the exchange algorithms in average time are precisely the reverse of the rankings in relative efficiency. A perfect ranking, in which the i^{th} algorithm always placed i^{th} in terms of execution time, would be indicated by a lower triangular matrix with all nonzero entries equal to 20. Tables 6(a) and (b) both very nearly assume this form. (Similar tables in terms of function evaluations were also constructed. Because these were essentially the same as Table 6(a) and (b), they have been omitted.)

In Tables 6(c) and (d), the relative efficiencies of the algorithms are compared. The $(i, j)^{th}$ entry in each of these tables gives the number of times that the i^{th} algorithm was strictly exceeded by the j^{th} algorithm in efficiency. Note that for these tables, the sum of the $(i, j)^{th}$ and $(j, i)^{th}$ entries is never equal to 20, as is the usual case in Tables 6(a) and (b). Algorithms often found designs which were equal, in terms of the

TABLE 5—*CP times and relative efficiencies of exchange algorithms averaged over all test cases.*

(a) Time

1.	Van Schalkwyk	9.95
2.	Wynn-Mitchell	11.41
3.	Mitchell	44.27
4.	Modified Fedorov	81.54
5.	Fedorov*	146.54

(b) Relative Efficiency

1.	Fedorov**	99.94
2.	Modified Fedorov	99.91
3.	Mitchell	99.44
4.	Wynn-Mitchell	98.81
5.	Van Schalkwyk	98.59

*Three runs terminated prior to convergence after 650 seconds.

**Based on 37 completed runs.

TABLE 6—*Pairwise comparisons of algorithms: execution times and relative efficiencies.*

(a) Execution Time* (Random Starting Designs)

	VS	W	M	MF	F
Van Schalkwyk (VS)	--	6	1	1	0
Wynn-Mitchell (W)	14	--	1	0	0
Mitchell (M)	19	19	--	3	2
Modified Fedorov (MF)	19	20	17	--	1
Fedorov (F)	20	20	18	19	--

(b) Execution Time* (Starting Designs Chosen from $S(\xi_D)$)

	K	VS	W	M	MF	F
Kiefer (K)	--	0	0	0	0	0
Van Schalkwyk (VS)	20	--	10	0	0	0
Wynn-Mitchell (W)	20	10	--	0	0	0
Mitchell (M)	20	20	20	--	4	3
Modified Fedorov (MF)	20	20	20	16	--	3
Fedorov (F)	20	20	20	17	16	--

(c) Relative Efficiency** (Random Starting Designs)

	MF	F	M	W	VS
Modified Fedorov (MF)	--	5	1	0	0
Fedorov (F)	7	--	2	1	1
Mitchell (M)	13	12	--	3	3
Wynn-Mitchell (W)	14	13	13	--	6
Van Schalkwyk (VS)	15	14	13	10	--

(d) Relative Efficiency** (Starting Designs Chosen from $S(\xi_D)$)

	F	MF	M	VS	K	W
Fedorov (F)	--	1	0	0	0	0
Modified Fedorov (MF)	4	--	1	0	1	0
Mitchell (M)	6	7	--	2	1	0
Van Schalkwyk (VS)	5	6	6	--	3	3
Kiefer (K)	8	9	9	6	--	6
Wynn-Mitchell (W)	6	8	7	5	3	--

*The (i,j)th entry gives the number of times that the ith algorithm exceeded the jth algorithm in execution time.

**The (i,j)th entry gives the number of times that the ith algorithm was strictly exceeded by the jth algorithm in relative efficiency.

determinant criterion, especially when the starting design was chosen from the support of the approximate *D*-optimal design. For example, from Table 6(d), the Van Schalkwyk and Fedorov algorithms resulted in the same determinant in 15 out of the 20 cases. As Tables 3(a) and (b) indicate, although the choice of starting design significantly affects required central processing time, it does not influence the relative performance of the algorithms. The results suggest, not surprisingly, that in general, the worse a starting design is in terms of relative efficiency the longer it will take an algorithm to converge to a "good" design. Also, few differences in ranking by efficiency can be attributed to the choice of starting design. As might have been expected, for the Wynn-Mitchell and Van Schalkwyk algorithms, better starting designs often yielded better resulting designs. However, with the Mitchell, modified Fedorov and Fedorov algorithms, use of random starting designs, more often than not, led to computed designs with higher efficiencies than those produced when the starting design was chosen from the support of the approximate *D*-optimal design.

5. CONCLUSIONS AND DISCUSSION

A fairly accurate summary of the results given in the previous section is provided by Table 5. Clearly, the better an algorithm is in terms of time, the worse it is in terms of *D*-efficiency, and vice versa. In general, those algorithms which are slower are so for a reason: they produce better designs. Therefore, the recommendations of this study may be given as follows:

1. In order to construct efficient designs inexpensively, either of the Wynn-Mitchell or Van Schalkwyk algorithms is satisfactory.

2. When highly efficient designs are required, use of the modified Fedorov algorithm is recommended.

The Fedorov algorithm, without modifications, may be prohibitively expensive to use. The modified version produced designs which were as efficient, overall, as those produced by the Fedorov procedure, requiring little over half of the computer time (see Table 5). The Mitchell algorithm is reasonable in terms of both time and relative efficiency. Large values of N may preclude use of even the modified Fedorov algorithm for the construction of highly efficient designs. In such a case, the Mitchell algorithm may represent the best choice.

Some particular limitations of the study need to be emphasized. The first is with regard to the optimization of the variance and delta functions. Since the combined grid-Powell search is not guaranteed to find a global maximum, the exchange algorithms compared are not strictly as proposed in the literature. The effect of this discrepancy is not clear. In general, of course, existing optimization routines cannot guarantee location of suprema and so "strict" application of the algorithms may not be possible with convex design spaces. Secondly, we again call attention to the inherent randomness in the results caused both by the choices of starting points for the Powell search and by the methods used for constructing starting designs. While this component does exist, the overall performances of the algorithms (i.e., over 40 test cases) were surprisingly consistent. Thus, the effect of this component is not considered to have influenced the rankings of the algorithms. Finally, we reiterate the fact that the study was designed only to describe the performance of the algorithms relative to each other. Nothing can be said, on the basis of this research, regarding an algorithm's ability to find a "best" design; that is, one with D-efficiency equal to 100%.

Two final comments must be made with regard to the Kiefer round-off procedure. First, for Type 1 models, the average rank of the Kiefer algorithm was 1.17, matched only by the modified Fedorov procedure. The result suggests that exact D-optimal designs for polynomial regression might often be supported on a subset of $S(\xi_D)$. Second, the Kiefer algorithm produced fairly efficient designs, using the least amount of computer time. Table 6(d) shows that, in one test case, the Kiefer algorithm found a better design than did the modified Fedorov procedure. As a result, if the approximate D-optimal design is known to the experimenter, little is lost in executing the Kiefer algorithm in addition to one of the exchange algorithms. Care must be taken in drawing comparisons between the Kiefer algorithm and any of the exchange algorithms. This is due to the fact that the design space $S(\xi_D)$ for the Kiefer procedure was finite, whereas the design space for each of the exchange algorithms was convex. Of interest would be a similar comparison of algorithms in which the design spaces would, in all cases, correspond to the supports of the approximate D-optimal designs.

6. ACKNOWLEDGEMENTS

R. D. Cook is a Visiting Staff Member of the Los Alamos Scientific Laboratory. This work was partially performed under the auspices of the United States Department of Energy (Contract W-7405-ENG.36).

The authors would like to thank the referees, the Associate Editor, and Kinley Larntz and S. Juan for helpful suggestions.

REFERENCES

ATKINSON, A. C. (1969). Constrained maximization and the design of experiments. *Technometrics, 11,* 616–618.

ATWOOD, C. L. (1970). Robust procedures for estimating polynomial regression. Tech. Report No. 143, School of Statistics, University of Minnesota.

BLOOM, T., Pfaffenberger, R., and Kochenberger, G. (1972). The efficient construction of optimal response surface designs based upon the $|\mathbf{X}'\mathbf{X}|$ criterion. Presented at the 1972 American Statistical Association Meeting, Montreal, Canada, August 14–18.

BOX, M. J. (1966). A comparison of several current optimization methods, and the use of transformations in constrained problems. *Computer J., 9,* 67–77.

BOX, M. J. and DRAPER, N. R. (1971). Factorial designs, the $|\mathbf{X}'\mathbf{X}|$ criterion, and some related matters. *Technometrics, 13,* 731–742.

CROWDER, H., DEMBO, R. S. and MULVEY, J. M. (1979). On reporting computational experiments with mathematical software. *ACM Trans. Math. Software 5, 2,* 193–203.

DRAPER, N. R. and ST. JOHN, R. C. (1977). A mixtures model with inverse terms. *Technometrics, 19,* 37–46.

FEDOROV, V. V. (1969). Theory of optimal experiments. Preprint No. 7 LSM, Izd-vo Moscow State University, Moscow, USSR.

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

HARTLEY, H. O. and RUUD, P. G. (1969). Computer optimization of second order response surface designs. In *Statistical Computing,* ed. by R. C. Milton and J. A. Nelder, pp. 441–462. New York: Academic Press.

KIEFER, J. (1971). The role of symmetry and approximation in exact design optimality. In *Statistical Decision Theory and Related Topics,* ed. by S. S. Gupta and J. Yachel, pp. 109–118. New York: Academic Press.

MAHONEY, F. (1970). The design of experiments for regression analysis by computer search procedures. Ph.D. thesis, Operations Research and Statistics, Rensselaer Polytechnic Institute.

MITCHELL, T. J. (1974). An algorithm for the construction of D-optimal experimental designs. *Technometrics, 16,* 203–211.

MITCHELL, T. J. and MILLER, F. L., JR. (1970). Use of design repair to construct designs for special linear models. *Math. Div. Ann. Progr. Rept. (ORNL-4661),* pp. 130–131. Oak Ridge National Laboratory, Oak Ridge, TN.

NACHTSHEIM, C. J. (1979). Contributions to optimal experimental design. Ph.D. thesis, Department of Applied Statistics, University of Minnesota.

NEUHARDT, J. B. and BRADLEY, H. E. (1971). Selection of

multi-factor experimental arrangements with resource constraints. *J. Amer. Statist. Assoc., 66,* 618–621.

POWELL, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Computer J., 7,* 155–162.

ST. JOHN, R. C. (1973). Models and designs for experiments with mixtures. Ph.D. thesis, Department of Statistics, University of Wisconsin–Madison.

ST. JOHN, R. C. and DRAPER, N. R. (1975). *D*-optimality for regression designs: A review. *Technometrics, 17,* 15–22.

ST. JOHN, R. C. and DRAPER, N. R. (1977). Designs in three and four components and mixture models with inverse terms. *Technometrics, 19,* 117–130.

VAN SCHALKWYK, D. J. (1971). On the design of mixture experiments. Ph.D. thesis, University of London.

WYNN, H. P. (1972). Results in the theory and construction of *D*-optimum experimental designs. *J. Roy. Statis. Soc., B-34,* 133–147.