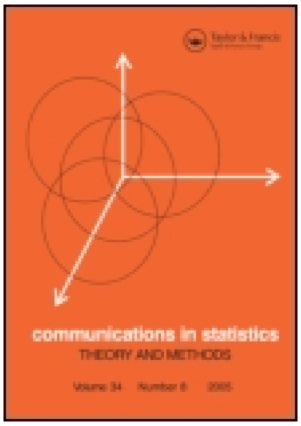


Communications in Statistics - Theory and Methods
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/lsta20

# Some iterative procedures for generating nonsingular optimal designs

Chien-Fu Wu [a]

[a] University of California, Berkeley

Version of record first published: 27 Jun 2007.

COMMUN. STATIST.-THEOR. METH., A7(14), 1399-1412 (1978)

# SOME ITERATIVE PROCEDURES FOR GENERATING NONSINGULAR OPTIMAL DESIGNS

Chien-Fu Wu

University of California, Berkeley

*Key and Words Phrases: optimal design algorithms; general equivalence theorem; D-optimality; A-optimality; vertex direction method; gradient projection method; conjugate gradient projection method.*

## ABSTRACT

The problem of optimal design algorithms is treated from the general viewpoint of optimization theory and algorithms. For designs of finite support, several iterative schemes are suggested, including vertex direction, gradient projection, normalized gradient projection and conjugate gradient projection. For an arbitrary design region, a general optimal design algorithm is proposed which incorporates the above algorithms for the finite support case and methods of changing design support. The numerical behavior of these algorithms is investigated in detail on six examples and some experiences with the algorithms are summarized in the last section.



## 1. INTRODUCTION

The problem of iterative generation of optimal design is investigated more systematically from the general viewpoint of optimization theory and algorithms. In Section 2 the connection of the optimal design problem on a finite design space with the convex programming problem is fully exploited in suggesting a class of algorithms for generating nonsingular optimal designs. In Section 3 the problem of changing the design support is discussed. The above two methods are combined to form a general optimal design algorithm for any arbitrary design space. Related theoretical results, especially the asymptotic convergence, are given in a separate paper (Wu, 1977). Several algorithms are compared on various examples in Section 4 and some numerical experiences are summarized in Section 5.

Consider the linear regression model $y = \theta^T x + \varepsilon$ where $\theta$ is $k \times 1$ vector, $x$ is from a compact set $X$ of $k \times 1$ vectors and $\varepsilon$ is random error. Errors corresponding to different observations are assumed to be uncorrelated, with equal variances and zero means. A design $\xi$ is defined to be a probability measure on $X$ and its corresponding information matrix $M(\xi)$ is defined to be $\int_X xx^T \xi(dx)$. Let $\mathcal{M}$ be the collection of $M(\xi)$'s for all probability measures $\xi$ on $X$. A design $\xi^*$ is called $\Phi$-optimal if it acnieves

$$\inf\{\Phi(M(\xi)): M(\xi) \in \mathcal{M}\} . \qquad (1)$$

Typical examples of $\Phi$ are $\Phi(M) = -\log\det M$ (D-optimality), $\Phi(M) = \mathrm{tr}(AM^{-p})$ for A positive definite and $p > 0$ (Kiefer's $\Phi_p$-optimality or L-optimality for $p = 1$) and $\Phi(M)$ = maximum eigenvalue of $M^{-1}$ (E-optimality).

Throughout the paper we assume that $\Phi$ is convex and bounded below on $\mathcal{M}$ and $\Phi$ is differentiable in a neighborhood of $\mathcal{M}^+$ in the space of nonnegative definite $k \times k$ matrices. $\mathcal{M}^+ = \{M: M \in \mathcal{M}$ and $\Phi(M) < \infty\}$.

When $\Phi$ is well-defined and differentiable, the $k \times k$ matrix $\nabla\Phi$ is defined as

$$(\nabla\Phi)_{ij} = \frac{\partial\Phi(M)}{\partial m_{ij}} .$$

The directional derivative

$$\frac{\partial}{\partial\alpha}\Phi((1-\alpha)M(\xi)+\alpha M(\xi'))\Big|_{\alpha=0^+} = \langle\nabla\Phi(M(\xi)),M(\xi'-\xi)\rangle$$
$$= \int_X x^T\nabla\Phi(M(\xi))x\xi'(dx)-tr(\nabla\Phi(M(\xi))M(\xi))$$

where $\langle A,B\rangle = tr(AB)$ since $\ell_2$-norm is used.

For convenience in presenting the algorithms, we shall use the following standard notation:

$$d(x,\xi) = -x^T\nabla\Phi(M(\xi))x\,,$$
$$\bar{d}(\xi) = \max_{x\in X} d(x,\xi)\,,$$
$$d^{\#}(\xi) = -tr(\nabla\Phi(M(\xi))M(\xi))\,.$$

In terms of the d-notation, the celebrated general equivalence theorem states that: $\xi^*$ is $\Phi$-optimal $\Leftrightarrow \xi^*\{x: d(x,\xi^*)=\bar{d}(\xi^*)\} = 1$ $\Leftrightarrow \bar{d}(\xi^*) = d^{\#}(\xi^*)$. (See Kiefer, 1974).

Motivated by the general equivalence theorem, Fedorov (1969) and Wynn (1970,1972) suggested the following iterative method for obtaining optimal designs:

$$\xi_{n+1} = (1-\alpha_n)\xi_n + \alpha_n\xi_{x_n}\,, \tag{2}$$

where $\xi_{x_n}$ is concentrated at $x_n$ with $d(x_n,\xi_n) = \bar{d}(\xi_n)$ and $\alpha_n$ is chosen to minimize $\Phi((1-\alpha)M(\xi_n)+\alpha M(\xi_{x_n}))$ over $0 \le \alpha \le 1$. Since all the extreme points of the convex set $\mathcal{M}$ are of the form $xx^T$, the Fedorov-Wynn algorithm only adjusts the design measure along the direction of the vertices of $\mathcal{M}$. Therefore we call the iterative procedure (3) the <u>vertex direction method</u>:

$$\xi_{n+1} = (1-\beta_n)\xi_n + \beta_n\xi_{y_n}\,, \tag{3}$$

where $y_n \in X$ and $\beta_n$ in $[-\xi_n(y_n)/1-\xi_n(y_n),1]$ can be chosen in various ways. Obviously the Fedorov-Wynn method is a special case of (3). Another special case of (3) is a modification of (2) due to Atwood (1973, suggestion 1 on p. 350). He compares the minimum $\Phi$-value according to (2) with

$$\min\{\Phi(M((1-\beta)\xi_n+\beta\xi_{y_n})): -\xi_n(y_n)/1-\xi_n(y_n) \le \beta \le 0\}\,, \tag{4}$$

where $d(y_n,\xi_n) = \min\{d(x,\xi_n): x \in X\}$ and then chooses the better one as the next iterative direction. Other modifications of (2) are suggested by Atwood (1973) and St. John and Draper (1975). For the D and L optimality criteria, closed forms of the minimizing $\alpha_n$ are available (Fedorov, 1969, p. 101, and Atwood, 1976, p. 1134). This saves a lot of effort in finding the minimum of $\Phi$ along the iterative direction. The method itself is indispensable in changing the design support. However, as we shall see later, it is not a very efficient minimization algorithm. In the framework of optimization theory, a method of Frank and Wolfe (1954) suggests minimizing the linear approximation of the $\Phi$ function. This reduces to (2) in the design situation. The less efficient Frank-Wolfe method has been superseded by other methods in the optimization literature. Hence we need to find some other algorithms which have better numerical properties.

Since problem (1) is apparently a convex programming problem over $M$, we certainly expect that our design problem will benefit from the vast area of optimization theory and algorithms. But several features of the design problem distinguish it from the typical constrained minimization problem:

(i) The optimal design $\xi^*$, instead of its information matrix $M(\xi^*)$, is the main concern. This excludes the use of duality theory, since we want to update the design at every iteration. This also explains why the iterative method (2) plays an important role in changing the design support.

(ii) The general equivalence theory of optimal design enables us to check optimality requirements analytically for some nice functions and regions. This is possible for other convex programming problems endowed with nice structures. Application of this approach to other fields appears very promising. Some work including Mehra (1974) has been done in optimal input control theory.

(iii) The value $\Phi$ may be $+\infty$ at some points of $M$, typically for singular $M(\xi)$. For example, $\Phi(M) = -\log\det M = \infty$ for $M$ singular.

This makes the convergence proofs for nonmonotone design sequences very hard. One example is the general step-length design algorithms of Wu and Wynn (1978).

## 2. SOME ITERATIVE SCHEMES FOR THE FINITE SUPPORT CASE

When $\mathcal{X}$ has finitely many points $\{x_i\}_{i=1}^n$, (1) can be rephrased as

$$\min\{\Phi(\sum_{i=1}^n \lambda_i x_i x_i^T): \sum_{i=1}^n \lambda_i = 1,\ \lambda_i \geq 0\}. \tag{5}$$

This is a typical constrained minimization problem with a nice constraint set, a simplex. The vast literature on this problem is a rich source for deriving useful algorithms for optimal design. Here we only consider several of them which are not too complicated to execute and to interpret in terms of the design language. Once the correspondence between (1) and (5) is established, it is always possible to produce algorithms other than what will be presented here.

Let $\xi_0 = (\lambda_i)_{i=1}^n$ be the initial design measure with all $\lambda_i$'s positive. The following class of iterative directions is considered:

$$h = (1^T\Lambda 1)(\Lambda d) - (1^T\Lambda d)(\Lambda 1)\ , \tag{6}$$

where $1^T = (1,\ldots,1)$, $d^T = (d_1,\ldots,d_n)$ with $d_i = d(x_i,\xi_0)$ and $\Lambda$ is a positive definite $n\times n$ matrix which may depend on $\xi_0$. Some reasons for considering (6) are:

(i) The direction h is a legitimate direction of iteration for probability measures, since $\Sigma_{i=1}^n h_i = (1^T\Lambda 1)(1^T\Lambda d)-(1^T\Lambda d)(1^T\Lambda 1) = 0.$

(ii) The derivative

$$\frac{\partial}{\partial\alpha}\Phi(M(\xi_0)+\alpha\sum_{i=1}^n h_i x_i x_i^T)\Big|_{\alpha=0^+} = \sum_{i=1}^n h_i x_i^T \nabla\Phi(M(\xi_0))x_i = -[(1^T\Lambda 1)(d^T\Lambda d)-(1^T\Lambda d)^2] \leq 0$$

with equality $\Leftrightarrow$ d is proportional to 1 (since $\Lambda$ is p.d.) $\Leftrightarrow$ $\xi_0$ is $\Phi$-optimal on $\mathcal{X}$ (from the general equivalence theorem). As long as the initial $\xi_0$ is not $\Phi$-optimal, iteration along the direction h will improve over $\xi_0$.

(iii) The $\Lambda$ matrix can be quite arbitrary. For different statistical or numerical reasons, different $\Lambda$'s can be considered. In fact, this class of methods is closely related to the optimal design algorithms proposed earlier by several authors.

Examples of h

1. Normalized gradient projection method: A reparameterized version of (5) is

$$\min\{\Phi(\sum_{i=1}^{n} u_i x_i x_i^T/|x_i|^2): \sum_{i=1}^{n} u_i/|x_i|^2 = 1,\ u_i \geq 0\}. \quad (7)$$

In the original problem (5) where $\lambda_i$ is the varying quantity, the Fedorov-Wynn method is just the steepest descent method. Since different direction $x_i$ has different length $|x_i|$, by considering $\lambda_i|x_i|^2$ as the varying quantity as in (7), the geometry of $M$ is taken care of. By projecting $(d_i/|x_i|^2)_{i=1}^n$, the negative gradient vector of $\Phi$ with respect to the $u_i$'s, onto the constraing subspace $\Sigma_{i=1}^n u_i/|x_i|^2 = 1$, we obtain the h direction in (6) with $\Lambda = \mathrm{diag}\{|x_1|^{-4},\ldots,|x_n|^{-4}\}$. Note that $|x_i| = 0$ is not included in (7).

2. Gradient projection method: By ignoring the effect of the length $|x_i|$, which amounts to projecting $(d_i)_{i=1}^n$ onto the constraint subspace $\Sigma_{i=1}^n \lambda_i = 1$, we obtain a simpler version of the previous method:

$$h = (d_i - d_.)_{i=1}^n, \quad d_. = n^{-1}\sum_{i=1}^{n} d_i \quad .$$

This corresponds to choosing $\Lambda = I$ in (6). Unlike the Fedorov-Wynn method and the Atwood modification, which only adjust the design measure along the maximizing or the minimizing d-direction, the present method adjusts all the design weights simultaneously according to the contributions of the d values. Compared with the other methods in the class (6), this is considerably simpler to execute and to understand.

3. $\Lambda = \mathrm{diag}\{\lambda_1,\ldots,\lambda_n\}$ in (6): The corresponding $h$ is

$$(\lambda_i d_i - (\sum_{j=1}^{n} \lambda_j d_j)\lambda_i)_{i=1}^{n} = (\lambda_i d_i - \lambda_i d^{\#})_{i=1}^{n} .$$

This is related to an algorithm suggested by Silvey, Titterington and Torsney (1977). Instead of choosing an iterative direction and then performing a line search along that direction, they suggest choosing the next design measure $\xi_1$ to be $(\lambda_i d(x_i,\xi_0)/d^{\#}(\xi_0))_{i=1}^{n}$. Since $d^{\#}(\xi_0) = \Sigma_{i=1}^{n}\lambda_i d(x_i,\xi_0)$, no line search is needed. Detailed numerical results of their algorithms are reported in their paper.

4. Conjugate gradient projection method: From the known fact in nonlinear programming that the conjugate gradient method is superior to the gradient method, the method in Example 2 can be improved by projecting the conjugate gradient direction of $\Phi$ onto the constraint subspace $\Sigma_{i=1}^{n}\lambda_i = 1$. More specifically, let $F(\lambda^{(j)}) = \Phi(\Sigma_{i=1}^{n}\lambda_i^{(j)}x_i x_i^T)$ and $\nabla F(\lambda) = (\partial F/\partial\lambda_i)_{i=1}^{n}$. The conjugate gradient directions are formed as

$$H_0 = -\nabla F(\lambda^{(0)}),$$
$$H_{j+1} = -\nabla F(\lambda^{(j+1)}) + c_j H_j, \quad j \geq 0.$$

In the numerical comparisons in Section 4, $c_j$ is chosen to be

$$b^{-1}|\nabla F(\lambda^{(j+1)})|^2/|H_j|^2 , \tag{8}$$

where $b$ is constant (the Fletcher-Rieves method), or to be

$$b_j^{-1}|\nabla F(\lambda^{(j+1)})|^2/|H_j|^2 , \tag{9}$$

where $b_j^{-1}$ decreases to zero as $\bar{d}(\xi_j)-d^{\#}(\xi_j)$ approaches zero (the adaptive conjugate gradient projection method). The h vector is obtained by projecting the $H_j$ vector onto $\Sigma_{i=1}^{n}\lambda_i = 1$ as before.

Other methods of solving (5) are available in the optimization literature. These include the Newton method, the Quasi-Newton method (especially the Fletcher-Powell method) and the parallel tangent method (PARTAN). The use of Newton method in design setting was proposed by Atwood (1976).

## 3. A GENERAL OPTIMAL DESIGN ALGORITHM

For an arbitrary $X$, a hybrid of the methods (2) and (6) is used. For design $\xi_j$ and $h(\xi_j)$ calculated according to (6), a line search for $\Phi(M(\xi_j+uh(\xi_j)))$ is performed on $[0,\bar{u}]$ where

$$\bar{u} = \min\{\lambda_i^{(j)}/|h_i(\xi_j)|: \quad h_i(\xi_j) < 0\} . \tag{10}$$

Note that $\xi_j + uh(\xi_j)$ is still a probability measure for $0 \leq u \leq \bar{u}$. If no optimal design exists on the support of $\xi_j$ and $\Phi(M(\xi_j))$ can not be improved very much on its current support, the effort of the above line search on $[0,\bar{u}]$ is almost futile. So when $\bar{u}\Sigma_{i=1}^{n} h_i(\xi_j)d(x_i,\xi_j)$, the directional derivative of $\Phi$ at $M(\xi_j)$ along $h(\xi_j)$, is small, the support of $\xi_j$ is augmented by the Fedorov-Wynn type algorithm and then method (6) is repeated; otherwise a line search on $[0,\bar{u}]$ is carried out. This is the idea behind the following general algorithm.

Algorithm: Choose $\varepsilon_0 > 0$, $0 < r \leq 1$.

step 0: $j = 0$, choose a $\xi_0$ with $\Phi(M(\xi_0)) < \infty$.

step 1: Compute $h$ and $\bar{u}$ of $\xi_j$ according to (6) and (10); if $\bar{u}\Sigma_{i=1}^{n} h_i d(x_i,\xi_j) \leq \varepsilon_0$, go to 3; else, go to 2.

step 2: Let $\xi_{j+1} = \xi_j + u_j h_j$ where $u_j$ minimizes $\Phi(M(\xi_j+uh_j))$ over $0 \leq u \leq \bar{u}$, $j = j+1$, go to 1.

step 3: If $\bar{d}(\xi_j) = d^{\#}(\xi_j)$, stop; else, choose an $y_j$ with $d(y_j,\xi_j) - d^{\#}(\xi_j) \geq r(\bar{d}(\xi_j)-d^{\#}(\xi_j))$, let $\xi_{j+1} = (1-\alpha_j)\xi_j + \alpha_j\xi_{y_j}$ with $\alpha_j$ minimizing $\Phi(M((1-\alpha)\xi_j + \alpha\xi_{y_j}))$ over $0 \leq \alpha \leq 1$, $j = j+1$, go to 1.

Remarks. 1. The case $r = 1$ corresponds to the Fedorov-Wynn algorithm. A similar way of choosing $y_j$ in step 3 has been suggested by Gustafson and Kortanek (1974) in their algorithm 1.

2. The value $\varepsilon_0$ in step 1 can be replaced by a sequence $\{\varepsilon_j\}$ which decreases to zero as $\bar{d}(\xi_j) - d^{\#}(\xi_j)$ goes to zero. The one-dimen-

sional minimization in steps 2 and 3 can be replaced by other existing line search methods. These modifications and their associated convergence proofs are given in Wu (1977).

## 4. NUMERICAL EXAMPLES

The following four methods are compared in detail on six examples with D-optimality and A-optimality ($\Phi(M) = \operatorname{tr} M^{-1}$).

I. Atwood's modification of (2) (see Section 1).

II. Gradient projection method of Section 2.

III. Conjugate gradient projection method of Section 2 with b in (8) equal to 2 and 20.

IV. Adaptive conjugate gradient projection method of Section 2 with $b_j$ in (9) equal to 64, 32, 16, 8, 4, 2, 1 according as $\bar{d}(\xi_j) - d^{\#}(\xi_j)$ is in $[0.001, 640^{-1})$, $[640^{-1}, 320^{-1})$, $[320^{-1}, 160^{-1})$, $[160^{-1}, 80^{-1})$, $[80^{-1}, 40^{-1})$, $[40^{-1}, 20^{-1})$, $[20^{-1}, \infty)$.

The conjugate gradient method is known to work better than the gradient method when the gradient vector does not go in the direction of the minimum point. Typically this happens at the earlier stages of the iterations where the local structures of $\Phi$ are very different from the local structure of $\Phi$ at the minimum. When the point is close to the minimum point, their local structures are quite similar and the current gradient direction should be trusted more than the previous direction. The adaptive choice of $b_j$ in IV seems to have both of these advantages. Our numerical experiences with b = 2 and 20 in III also verify the above phenomenon.

Example 1. $\theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2$, $(x_1, x_2) \in X = \{v_1, v_2, v_3, v_4\} = \{(-1,-1),(-1,1),(1,-1),(2,2)\}$, the quadrilateral with these four points as vertices. This example was used by several authors in comparing their algorithms (Wynn,1970; Atwood,1973, 1976; Titterington,1976).

Example 2. As in Example 1, but with $v_4 = (2,3)$.

<u>Example 3</u>. As in Example 1, but with $v_4 = (-1,-2)$.

<u>Example 4</u>. $\theta^T x = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$, $(x_1,x_2,x_3) \in X$ $= \{(1,-1,-1),(-1,1,-1),(-1,-1,-1),(2,2,-1),(1,-1,1),(-1.5,1,1),$ $(-1,-1,2)\}$.

<u>Example 5</u>. As in Example 4, but with the additional point (1,1.5,1) which however has zero weight at the optimal design.

<u>Example 6</u>. $\theta^T x = \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_1 x_2 + \theta_5 x_1 x_3 + \theta_6 x_2 x_3$, $x_1, x_2, x_3 \geq 0$ and $x_1 + x_2 + x_3 = 1$. This example was considered by Kiefer (1975) where he showed that an optimal design is supported at the barycenters of the simplex: (1,0,0), (0,1,0), (0,0,1), $(\frac{1}{2},\frac{1}{2},0)$, $(\frac{1}{2},0,\frac{1}{2})$, $(0,\frac{1}{2},\frac{1}{2})$, $(\frac{1}{3},\frac{1}{3},\frac{1}{3})$.

Since the first five examples are linear regression models and both the D and A criteria are convex functions, the optimal designs are supported on the vertices of $X$. Therefore the considerations of optimal designs can be restricted to a finite number of points. For the first three examples the initial design is chosen to be $(\frac{1}{3},\frac{1}{3},\frac{1}{3},0)$ in accordance with the choice of the previous authors. For the next three, all the algorithms start with the uniform measure.

Some other methods are also tried on Examples 1 to 3 with D-optimality. For Examples 1 and 3, the Fedorov-Wynn method takes 109 and 66 iterations to achieve $\bar{d}(\xi_j) - 3 \leq 0.001$. For Example 2, it does not achieve $\bar{d}(\xi_j) - 3 \leq 0.001$ even at the 160th iteration ($\bar{d}(\xi_{160}) = 3.00492$). Compared with the methods listed in table I, this is very inefficient. The simple modification due to Atwood (method I) performs much better than the original F-W method. Therefore it is not considered in the later comparisons on more complicated examples. The normalized gradient projection method of Section 2 is also tried on Examples 1 to 3 with D-optimality. The required number of iterations to achieve $\bar{d}(\xi_j) - 3 \leq 0.001$ ranges from 6 to 14, slightly more than the four methods in table I.

For Examples 1 to 5 with D-optimality, the numbers of iterations required to achieve $\bar{d}(\xi_j) - d^{\#}(\xi_j) \leq 0.1$, 0.01, 0.001 are listed in Table I. For Examples 4 to 6 with A-optimality, similar values are listed in Table II. Note that $d(x,\xi) = x^T M^{-1}(\xi)x$, $d^{\#}(\xi) = k$ for D-criterion and $d(x,\xi) = x^T M^{-2}(\xi)x$, $d^{\#}(\xi) = \text{tr } M^{-1}(\xi)$ for A-criterion.

## 5. DISCUSSION AND SUMMARY

Our numerical experiences are summarized as follows.

(i) It is clear that the adaptive conjugate gradient projection method is the fastest and the Atwood method is the slowest. The conjugate gradient projection method slightly improves

TABLE I

D-optimality

| examples \ methods | I | II | III (b = 2) | III (b = 20) | IV |
|---|---|---|---|---|---|
| 1 | 3,5,7 | 2,3,3 | 2,3,4 | 2,3,3 | 2,3,3 |
| 2 | 5,6,7 | 3,6,9 | 3,5,6 | 3,6,9 | 3,4,6 |
| 3 | 4,7,8 | 3,4,6 | 3,4,5 | 3,4,6 | 3,4,4 |
| 4 | 12,37,64 | 5,15,27 | 4,9,15 | 5,15,25 | 4,5,15 |
| 5 | 6,59,>900 | 6,22,32 | 6,14,63 | 6,16,32 | 4,8,13 |

TABLE II

A-optimality

| examples \ methods | I | II | III (b = 2) | III (b = 20) | IV |
|---|---|---|---|---|---|
| 4 | 16,35,51 | 8,18,28 | 4,12,22 | 7,17,28 | 5,11,19 |
| 5 | 6,63,141 | 4,28,58 | 3,29,55 | 4,28,58 | 5,27,57 |
| 6 | 19,24,30 | 12,16,21 | 10,14,19 | 8,11,14 | 9,11,16 |

over the gradient projection method. But on simple design problems like Examples 1 to 3 all these methods are practically the same. The quadratic method of Atwood (1976) is expected to perform as well as the adaptive method. The performance of method 3 of Section 2 should be comparable to the gradient projection method (normalized or not), although this has never been tested on the above examples.

(ii) There are two numerical advantages in using the vertex direction method for the D and A cases. First, the $\alpha^*$ which minimizes $\Phi(M((1-\alpha)\xi_n + \alpha\xi_{x_n}))$ over an interval is obtainable without any line search (also see Section 1). For each iteration this saves at least 10 evaluations of det M or tr $M^{-1}$. The second advantage is that $M^{-1}(\xi_{n+1})$ can be obtained simply from $M^{-1}(\xi_n)$ through an iterative formula (Fedorov, 1969, p. 106). For the Atwood method these two advantages will probably more than compensate for its slow convergence. However, the numerical pros and cons of various methods depend also on how cheaply det M and tr $M^{-1}$ can be evaluated without inverting M. The quadratic method of Atwood involves inverting a Hessian matrix whose order is the same as the number of support points. As the support size increases, the computation becomes formidable. Atwood later suggested using the diagonalized Hessian matrix to resolve this numerical problem. The efficiencies of both the conjugate gradient projection method and its adaptive modification depend on the choices of the mixing coefficients b in (8) and $b_j$ in (9). In practice they are selected more or less on an ad hoc basis. Our choices of b and $b_j$ seem to work very well.

(iii) The Atwood method has been treated merely as a simple modification of the Fedorov-Wynn method and hence has received very little attention. In fact, in all the examples we have considered, more of the iterations of the Atwood method involve subtracting rather than adding mass. In Example 4 with D-optimality, out of the 64 iterations, 52 involve subtracting mass and only 12 involve adding mass. The figure for Example 6 with A-optimality is even more striking. None of the 30 iterations involve adding mass. This may explain why the Atwood "amendment" of the Fedorov-Wynn method is far superior. Together with remark (ii), this suggests the use of Atwood method for design problems of small or moderate size.

## ACKNOWLEDGEMENTS

This paper was prepared with the support of the Office of Naval Research Contract No. N00014-75-C-0444/NR 042-036. It is based on part of the author's Ph.D. dissertation written at the University of California, Berkeley under the supervision of Professor Peter J. Bickel. The author would like to thank Professors P. J. Bickel, J. Kiefer, H. P. Wynn for encouragement and help, and Mr. Paul Wang for programming assistance.

## BIBLIOGRAPHY

Atwood, C.L. (1973). Sequences converging to D-optimal designs of experiments. *Ann. Statist. 1*, 342-52.

Atwood, C.L. (1976). Convergent design sequences for sufficiently regular optimality criteria. *Ann. Statist. 4*, 1124-38.

Fedorov, V.V. (1969, Transl. 1972). *Theory of Optimal Experiments*. New York: Academic Press.

Frank, M. & Wolfe, P. (1956). An algorithm for quadratic programming. *Naval Res. Logist. Quart. 3*, 95-110.

Gustafson, S.Å. & Kortanek, K.O. (1974). Constructing optimal experimental designs using convexity. NOPA Series Rep.12, Carnegie-Mellon University, Pittsburgh, Pennsylvania.

Kiefer, J. (1974). General equivalence theory for optimal designs (approximate theory). *Ann. Statist. 2*, 849-79.

Kiefer, J. (1975). Optimal design: variation in structure and performance under change of criterion. *Biometrika 62*, 277-88.

Mehra, R. (1974). Optimal input signals for parameter estimation in dynamic systems--survey and new results. *IEEE Trans. Automat. Contr. AC-19*, 753-68.

St. John, R.C. & Draper, N.R. (1975). D-optimality for regression designs: a review. *Technometrics 17*, 15-23.

Silvey, S.D., Titterington, D.M. & Torsney, B. (1978). An algorithm for optimal designs on a finite design space. *Commun. Statist. A.*

Titterington, D.M. (1976). Algorithms for computing D-optimal designs on a finite design space. *Proc. 1976 Conf. on Information Sciences and Systems*, Johns Hopkins University, 213-16.

Wu, C.F. (1977). Some algorithmic aspects of the theory of optimal designs. Tech. Report No. 499, Dept. of Statistics, University of Wisconsin, Madison.

Wu, C.F. & Wynn, H.P. (1978). The convergence of general step-length algorithms for regular optimum design criteria. *Ann. Statist.* To appear.

Wynn, H.P. (1970). The sequential generation of D-optimum experimental designs. *Ann. Math. Statist. 41*, 1655-64.

Wynn, H.P. (1972). Results in the theory and construction of D-optimum experimental designs. *J. R. Statist. Soc. B 34*, 133-47.

*Received June, 1977.*

*Refereed anonymously.*