

# Branch-and-Bound Search for Experimental Designs Based on $D$ Optimality and Other Criteria

William J. Welch

Department of Mathematics  
Imperial College,  
London SW7 2BZ

This article presents a branch-and-bound algorithm that constructs a catalog of all  $D$ -optimal  $n$ -point designs for specified design region, linear model, and number of observations,  $n$ . While the primary design criterion is  $D$  optimality, the algorithm may also be used to find designs performing well by other secondary criteria, if a small sacrifice in efficiency as measured by  $D$  optimality is accepted. Finally, some designs are supplied for a quadratic response surface model.

KEY WORDS:  $D$ -optimal; Design; Branch and bound; Augmentation; Quadratic response surface.

## 1. INTRODUCTION

We are concerned with the problem of choosing  $n$  experimental settings or points  $x_{(i)}$  ( $i = 1, \dots, n$ ), possibly vector-valued and not necessarily distinct, from a specified design region  $\chi$  at which to take observations  $Y_{(i)}$  ( $i = 1, \dots, n$ ). In addition, our interest will center upon the situation in which observations are expensive and, hence,  $n$  is fixed at a value small enough to render the approximate design theory of Kiefer and Wolfowitz (1959) inappropriate. We assume the linear model

$$Y_{(i)} = \mathbf{f}(x_{(i)})^T \boldsymbol{\beta} + \varepsilon_{(i)}, \quad i = 1, \dots, n,$$

where  $\mathbf{f}$  is a  $k \times 1$  vector of given linearly independent functions on  $\chi$  ( $k \leq n$ ),  $\boldsymbol{\beta}$  is a  $k \times 1$  vector of unknown parameters, and  $\varepsilon_{(i)}$  is a random variable for error. It is further assumed that the errors  $\varepsilon_{(i)}$  ( $i = 1, \dots, n$ ) are independently distributed, each with mean zero and variance  $\sigma^2$ .

Attention will be restricted to cases in which  $\chi$  is a design region of  $r$  candidate sites  $x_j$  ( $j = 1, \dots, r$ ), thereby excluding continuous regions, since (a) certain computational advantages ensue, and (b) the experimental settings may be qualitative or, if quantitative, the experimenter may wish to limit each factor to only a few levels.

Adapting the notation of Kiefer and Wolfowitz, we represent an  $n$ -point design assigning  $n_j$  points to site  $x_j$  ( $j = 1, \dots, r$ ;  $\sum_{j=1}^r n_j = n$ ) by an  $r \times 1$  vector

$\mathbf{p}_n = (p_1, \dots, p_r)$  where  $p_j = n_j n^{-1}$ . For the design  $\mathbf{p}_n$  define the  $k \times k$  matrix  $\mathbf{M}(\mathbf{p}_n) = \sum_{j=1}^r p_j \mathbf{f}(x_j) \mathbf{f}(x_j)^T$ , a normalized (for  $n$ ) version of the familiar  $\mathbf{X}^T \mathbf{X}$  matrix. Our primary criterion for choosing the design will be that of  $D$  optimality, proposed by Wald (1943) but given the current nomenclature by Kiefer (1958). A  $D$ -optimal  $n$ -point design or, for brevity,  $D_n$ -optimal design,  $\mathbf{p}_n^*$  satisfies

$\mathbf{p}_n^*$  maximizes  $\det\{\mathbf{M}(\mathbf{p}_n)\}$  subject to

$$0 \leq p_j \leq 1, \quad j = 1, \dots, r \quad (1)$$

$$\sum_{j=1}^r p_j = 1, \quad (2)$$

$$p_j = n_j n^{-1}, \quad n_j \text{ an integer, } j = 1, \dots, r. \quad (3)$$

For designs  $\mathbf{p}_n$  where  $\mathbf{M}(\mathbf{p}_n)$  is nonsingular and hence the parameters  $\boldsymbol{\beta}$  are estimable, we also need to define the quantities

$$d(x_j, x_{j'}, \mathbf{p}_n) = \mathbf{f}(x_j)^T \mathbf{M}^{-1}(\mathbf{p}_n) \mathbf{f}(x_{j'}), \quad j, j' = 1, \dots, r$$

and

$$d(x_j, \mathbf{p}_n) = d(x_j, x_j, \mathbf{p}_n), \quad j = 1, \dots, r.$$

To aid interpretation of these definitions, note that  $\sigma^2 n^{-1} \mathbf{M}^{-1}(\mathbf{p}_n)$  is the covariance matrix of the least squares estimators  $\hat{\boldsymbol{\beta}}$  for the design  $\mathbf{p}_n$  and  $\sigma^2 n^{-1} d(x_j, x_{j'}, \mathbf{p}_n)$  represents the covariance between the estimated responses  $\mathbf{f}(x_j)^T \hat{\boldsymbol{\beta}}$  and  $\mathbf{f}(x_{j'})^T \hat{\boldsymbol{\beta}}$  at  $x_j$  and  $x_{j'}$  ( $j, j' = 1, \dots, r$ ).

With  $n$  design points and  $r$  candidate sites there are  $\binom{r+n-1}{n}$  possible designs (though often some will leave the parameters inestimable) and, hence, one way to construct a  $D_n$ -optimal design would be to compute and compare all  $\binom{r+n-1}{n}$  determinants  $\det\{\mathbf{M}(\mathbf{p}_n)\}$ . Clearly, though, even for moderate  $n$  and  $r$ , an exhaustive search is computationally infeasible. Box and Draper (1971) perform such a complete enumeration for small factorial experiments, but for larger problems they restrict attention to cube and star point designs. An integer programming approach for constrained factorial designs is discussed by Neuhardt and Bradley (1971).

If the  $n$ -point condition (3) is removed from the optimization problem discussed previously, the maximization is computationally much easier: the algorithms of, for example, Wynn (1970) or Fedorov (1972, pp. 97–104) may be applied, followed by the “rounding” algorithm of Fedorov (p. 157) to recover an  $n$ -point design. A  $D_n$ -optimal design is not guaranteed, however, and for small  $n$  (near  $k$ ) performance may be poor.

The exchange algorithms of Fedorov (1972, pp. 160–165), Wynn (1972), and Mitchell (1974) are all sequential: a sequence of  $n$ -point designs with nondecreasing determinants is generated but, while computationally inexpensive compared with exhaustive search, convergence, though assured, may be to a local maximum and a  $D_n$ -optimal design is again not guaranteed. The balanced array method of Mitchell and Bayne (1978) for fractions of three-level factorial arrangements may also generate local maxima. Cook and Nachtsheim (1980) review and compare algorithms for constructing  $D_n$ -optimal designs.

## 2. A BRANCH-AND-BOUND ALGORITHM

The concept of branch and bound was popularized largely by the algorithm of Little, Murty, Sweeney, and Karel (1963) for the traveling salesman problem. We now discuss a branch-and-bound algorithm for the  $D_n$ -optimal design problem, with an extension considering other criteria in conjunction with  $D$  optimality.

### 2.1 The Binary Tree (branching) Structure

The optimization problem to maximize  $\det\{\mathbf{M}(\mathbf{p}_n)\}$  subject to (1), (2), and (3) may be generalized if the constraint (1) is replaced by

$$l_j \leq p_j \leq u_j, j = 1, \dots, r, \quad (4)$$

where  $l_j$  and  $u_j$  are specified constants such that  $0 \leq l_j \leq u_j \leq 1$  ( $j = 1, \dots, r$ ). Hereafter, the maximization subject to (2), (3), and (4) for specified  $\mathbf{l}$  and  $\mathbf{u}$  will be called a node and for the original maximization, or root node, we merely write  $l_j = 0, u_j = 1$  ( $j = 1, \dots, r$ ). Rather than solve the root node direc-

tly, though, we may solve two subproblems, or descendant nodes, by choosing the constants  $\mathbf{l}$  and  $\mathbf{u}$  of each descendant to divide the set of all possible  $n$ -point designs into two partitions. Comparison of the two solutions of the descendants trivially solves the root.

The partitioning or branching process may be continued recursively: for every node, including the root, one of two conditions holds. Either

1. the constraints (2), (3), and (4) of the node allow one and only one  $n$ -point design and the node maximization immediately follows, or

2. the constraints admit more than one  $n$ -point design, whereupon we select integers  $j_0(\mathbf{l}, \mathbf{u})$  ( $1 \leq j_0(\mathbf{l}, \mathbf{u}) \leq r$ ) and  $n_0(\mathbf{l}, \mathbf{u})$  and create two descendant nodes. The descendants partition the  $n$ -point designs allowed by the current node into those with at least  $n_0$  design points  $x_{j_0}$  and those with less than  $n_0$  points  $x_{j_0}$  (we omit the dependence of  $j_0$  and  $n_0$  on  $\mathbf{l}$  and  $\mathbf{u}$  for typographical brevity). To effect a division into non-empty partitions we require that

$$1 + l_{j_0} n \leq n_0 \leq u_{j_0} n,$$

$$\sum_{\substack{j=1 \\ j \neq j_0}}^r l_j + n_0 n^{-1} \leq 1,$$

and

$$\sum_{\substack{j=1 \\ j \neq j_0}}^r u_j + (n_0 - 1)n^{-1} \geq 1.$$

Finally, we update the constraints (4): the descendant nodes take the values of  $\mathbf{l}$  and  $\mathbf{u}$  of the current node, except that for the first descendant  $l_{j_0} = n_0 n^{-1}$  and for the second  $u_{j_0} = (n_0 - 1)n^{-1}$ .

Thus, every node has either zero or two descendant nodes and a binary tree is generated, with the  $\binom{r+n-1}{n}$   $n$ -point designs located at the extremes. Comparison of their determinants clearly guarantees a  $D_n$ -optimal design, but this is essentially an exhaustive search; indeed, it is an inefficient method of exhaustive search. By introducing bounds, however, much of the tree need not be explicitly evaluated and major computational improvements ensue.

### 2.2 Bounds

Consider a node where the constraints (2), (3), and (4) allow more than one  $n$ -point design. We now give two upper bounds for the node maximization.

For the first bound, we need to define

$$n^{(a)} = n \sum_{j=1}^r l_j,$$

$$\mathbf{M}(\mathbf{l}, \alpha) = \sum_{j=1}^r \left( l_j + \frac{\alpha}{nr} \right) \mathbf{f}(x_j) \mathbf{f}(x_j)^T, \quad \alpha > 0 \quad (5)$$

and

$$d(x_j, \mathbf{l}, \alpha) = \mathbf{f}(x_j)^T \mathbf{M}^{-1}(\mathbf{l}, \alpha) \mathbf{f}(x_j),$$

where (5) ensures that the  $k \times k$  matrix  $\mathbf{M}(\mathbf{l}, \alpha)$  is positive definite, following Mitchell (1974). For small  $\alpha > 0$ , if  $\mathbf{M}(\mathbf{l}, 0)$  is already positive definite, then the proportional error in using  $\det\{\mathbf{M}(\mathbf{l}, \alpha)\}$  below, instead of  $\det\{\mathbf{M}(\mathbf{l}, 0)\}$ , is approximately  $\alpha$ . In later computations  $\alpha$  is set at .001. The integer  $n^{(a)}$  is the number of points assigned by  $\mathbf{l}$ ; therefore, any design  $\mathbf{p}_n$  satisfying the node constraints is composed of the  $n^{(a)}$  assigned points plus any further  $n - n^{(a)}$  points allowed by the constraints. Let these further points be  $x_{(i)}$  ( $i = n^{(a)} + 1, \dots, n$ ). Now

$$\begin{aligned} \det\{\mathbf{M}(\mathbf{p}_n)\} &= \det\{\mathbf{M}(\mathbf{l}, 0) + n^{-1} \mathbf{F} \mathbf{F}^T\} \\ &\leq \det\{\mathbf{M}(\mathbf{l}, \alpha) + n^{-1} \mathbf{F} \mathbf{F}^T\} \end{aligned}$$

and  $\mathbf{F}$  is the  $k \times (n - n^{(a)})$  matrix  $[\mathbf{f}(x_{(n^{(a)}+1)}), \dots, \mathbf{f}(x_{(n)})]$ . Applying Lemma 2.5.1 of Fedorov (1972, pp. 99-100) and the Hadamard inequality for positive definite matrices, it follows that

$$\begin{aligned} \det\{\mathbf{M}(\mathbf{p}_n)\} &\leq \det\{\mathbf{M}(\mathbf{l}, \alpha)\} \det\{I + n^{-1} \mathbf{F}^T \mathbf{M}^{-1}(\mathbf{l}, \alpha) \mathbf{F}\} \\ &\leq \det\{\mathbf{M}(\mathbf{l}, \alpha)\} \prod_{i=n^{(a)}+1}^n \{1 + d(x_{(i)}, \mathbf{l}, \alpha) n^{-1}\} \\ &\leq \det\{\mathbf{M}(\mathbf{l}, \alpha)\} \{1 + \bar{d}(\mathbf{l}, \alpha) n^{-1}\}^{n-n^{(a)}}, \end{aligned} \tag{6}$$

where

$$\bar{d}(\mathbf{l}, \alpha) = \max_{\substack{j=1, \dots, r \\ l_j < u_j}} d(x_j, \mathbf{l}, \alpha)$$

and  $I$  is the  $(n - n^{(a)}) \times (n - n^{(a)})$  identity matrix. The bound on the maximization provided by the right-hand side of (6) is denoted by  $b_1(\mathbf{l}, \mathbf{u})$ . Note that if terms involving  $\alpha$  are neglected, then (6) is an equality when  $n - n^{(a)} = 1$  and the node maximization follows.

The second upper bound on the node maximization is obtained by removing the  $n$ -point condition (3) and so relaxing the optimization problem (the relaxed problem cannot yield a lower maximum). The constraints (2) and (4) in isolation define a class of approximate designs, approximate in the sense that  $p_j$  ( $j = 1, \dots, r$ ) need not correspond to an integer number of observations. Theorem 1 below, an adaptation of Theorem 2.1 of Wynn (1977), suggests a computational method for the relaxed maximization.

*Theorem 1.* The following two conditions on  $\mathbf{p}^*$  are equivalent. (a)  $\mathbf{p}^*$  maximizes  $\det\{\mathbf{M}(\mathbf{p})\}$  subject to (2) and (4). (b)  $\mathbf{p}^*$  satisfies (2) and (4) and for all  $h$  such that  $p_h^* < u_h$  ( $h = 1, \dots, r$ ) and for all  $i$  such that  $p_i^* > l_i$  ( $i = 1, \dots, r$ ),  $d(x_h, \mathbf{p}^*) \leq d(x_i, \mathbf{p}^*)$ . The proof may be derived from that of Wynn and is omitted.

From an arbitrary approximate design  $\mathbf{p}^{(1)}$  sat-

isfying (2) and (4) with  $\mathbf{M}(\mathbf{p}^{(1)})$  nonsingular we iteratively generate a sequence of designs  $\mathbf{p}^{(s)}$  ( $s = 1, 2, \dots$ ). At iteration  $s$  let  $h$  and  $i$  satisfy

$$d(x_h, \mathbf{p}^{(s)}) = \max_{\substack{j=1, \dots, r \\ p_j^{(s)} < u_j}} d(x_j, \mathbf{p}^{(s)})$$

and

$$d(x_i, \mathbf{p}^{(s)}) = \min_{\substack{j=1, \dots, r \\ p_j^{(s)} > l_j}} d(x_j, \mathbf{p}^{(s)}).$$

Then  $\mathbf{p}^{(s+1)}$  is derived from  $\mathbf{p}^{(s)}$  as

$$p_j^{(s+1)} = \begin{cases} p_j^{(s)}, & j = 1, \dots, r; j \neq h, i \\ p_j^{(s)} + \delta^{(s)}, & j = h \\ p_j^{(s)} - \delta^{(s)}, & j = i. \end{cases}$$

It may be shown that

$$\det\{\mathbf{M}(\mathbf{p}^{(s+1)})\} = (1 + \Delta) \det\{\mathbf{M}(\mathbf{p}^{(s)})\}$$

where

$$\begin{aligned} \Delta &= \delta^{(s)} \{d(x_h, \mathbf{p}^{(s)}) - d(x_i, \mathbf{p}^{(s)})\} \\ &\quad - (\delta^{(s)})^2 \{d(x_h, \mathbf{p}^{(s)}) d(x_i, \mathbf{p}^{(s)}) - d^2(x_h, x_i, \mathbf{p}^{(s)})\}. \end{aligned}$$

By differentiating  $\Delta$  with respect to  $\delta^{(s)}$  and equating with zero we find that to maximize  $\det\{\mathbf{M}(\mathbf{p}^{(s+1)})\} / \det\{\mathbf{M}(\mathbf{p}^{(s)})\}$  and satisfy the constraints imposed by  $u_h$  and  $l_i$

$$\delta^{(s)} = \min \left[ u_h - p_h^{(s)}, p_i^{(s)} - l_i, \frac{d(x_h, \mathbf{p}^{(s)}) - d(x_i, \mathbf{p}^{(s)})}{2\{d(x_h, \mathbf{p}^{(s)}) d(x_i, \mathbf{p}^{(s)}) - d^2(x_h, x_i, \mathbf{p}^{(s)})\}} \right].$$

Clearly,  $\det\{\mathbf{M}(\mathbf{p}^{(s+1)})\} \geq \det\{\mathbf{M}(\mathbf{p}^{(s)})\}$  since equality can be achieved if  $\delta^{(s)} = 0$ . Furthermore, by Theorem 1, the monotonic nondecreasing sequence of determinants  $\det\{\mathbf{M}(\mathbf{p}^{(s)})\}$  converges to  $\det\{\mathbf{M}(\mathbf{p}^*)\}$ , an upper bound on the node maximization also incorporating (3), which we denote by  $b_2(\mathbf{l}, \mathbf{u})$ .

For a node in the binary tree where the constraints allow more than one  $n$ -point design, we seek to avoid creating the descendants by employing the bounds  $b_1(\mathbf{l}, \mathbf{u})$  and  $b_2(\mathbf{l}, \mathbf{u})$ . Let  $D^*$  represent the maximum determinant  $\det\{\mathbf{M}(\mathbf{p}_n)\}$  of those  $n$ -point designs encountered by the algorithm prior to consideration of the current node. If

$$\min\{b_1(\mathbf{l}, \mathbf{u}), b_2(\mathbf{l}, \mathbf{u})\} < D^* \tag{7}$$

holds, then all  $n$ -point designs allowed by the node constraints have a determinant inferior to  $D^*$ . Therefore, we need not consider any descendants of the current node and do not proceed with the branching operation.

The bound  $b_2(\mathbf{l}, \mathbf{u})$  is computed as the limit of a sequence, but in practice we are only concerned whether  $b_2(\mathbf{l}, \mathbf{u}) < D^*$  or  $b_2(\mathbf{l}, \mathbf{u}) \geq D^*$  [assume that

$b_1(\mathbf{l}, \mathbf{u}) \geq D^*$ . At iteration  $s$ , if  $\det\{\mathbf{M}(\mathbf{p}^{(s)})\} \geq D^*$  then, clearly, the second case is true and the sequence is terminated. However, if  $\det\{\mathbf{M}(\mathbf{p}^{(s)})\} < D^*$  then, rather than invoke an arbitrary stopping rule, we employ Theorem 2 below, the proof of which closely follows Kiefer (1961).

*Theorem 2.* For an approximate design  $\mathbf{p}$  satisfying (2) and (4) the inequality

$$b_2(\mathbf{l}, \mathbf{u}) < \det\{\mathbf{M}(\mathbf{p})\} \exp\{d^*(\mathbf{p}) - k\}$$

holds, where

$$d^*(\mathbf{p}) = \max_{\mathbf{p}' \text{ satisfies (2) and (4)}} \sum_{j=1}^r p'_j d(x_j, \mathbf{p}).$$

To compute the vector  $\mathbf{p}'$  providing  $d^*(\mathbf{p})$ , we initially set  $p'_j + l_j$  ( $j = 1, \dots, r$ ) to satisfy the lower bounds in (4). The remaining  $1 - \sum_{j=1}^r l_j$  is then assigned by increasing the elements  $p'_j$  subject to the upper bounds in (4) in order of decreasing  $d(x_j, \mathbf{p})$  ( $j = 1, \dots, r$ ) until  $\sum_{j=1}^r p'_j = 1$ . The implication of Theorem 2 is that when  $\det\{\mathbf{M}(\mathbf{p}^{(s)})\} < D^*$ , if

$$\det\{\mathbf{M}(\mathbf{p}^{(s)})\} \exp\{d^*(\mathbf{p}^{(s)}) - k\} < D^*,$$

then  $b_2(\mathbf{l}, \mathbf{u}) < D^*$  and the sequence may be terminated, whereas if

$$\det\{\mathbf{M}(\mathbf{p}^{(s)})\} \exp\{d^*(\mathbf{p}^{(s)}) - k\} \geq D^*,$$

we continue iterating.

### 2.3 A Strategy for Branching

The strategy proposed for choosing  $j_0$  and  $n_0$  in the branching operation is based on empirical studies only. We use the heuristics

$$d(x_{j_0}, \mathbf{l}, \alpha) = \max_{\substack{j=1, \dots, r \\ l_j < u_j}} d(x_j, \mathbf{l}, \alpha) \quad (8)$$

and  $n_0 = l_{j_0} n + 1$  to select  $j_0$  and  $n_0$ . Thus, we attempt to branch such that the descendant with increased lower constraint  $l_{j_0}$  is favored relative to the descendant with decreased upper constraint  $u_{j_0}$ ; similar strategies have been found effective in integer programming (see, for example, Garfinkel and Nemhauser 1972). The rule for selecting  $n_0$  assumes that we are interested in design problems in which  $n - k$  is small.

If branching takes place, the next node to be evaluated is always the descendant with increased lower constraint  $l_{j_0}$ , and the other descendant is placed on the top of a stack of nodes awaiting attention. Branching does not take place if the bound (7) applies or the node optimization is solved (and  $D^*$  is updated if necessary), a case arising when one and only one  $n$ -point design is defined by the node constraints or  $n - n^{(a)} = 1$ . When nodes with no descendants occur, the next node to be processed is removed from the top of the stack. The algorithm commences with an empty stack,  $D^*$  is initialized at zero, and the current node of interest is the root. Termination occurs when a node with no descendants is encountered and the stack is empty, and the design or designs associated with  $D^*$  at termination are the  $D_n$ -optimal designs.

### 2.4 An Example

Let  $n = 3$  and let the model be  $E(Y_{(i)}) = \beta_1 + \beta_2 x_{(i)}$  ( $i = 1, 2, 3$ ) with design region comprising the  $r = 3$  candidates  $x_1 = -1$ ,  $x_2 = 0$ , and  $x_3 = 1$ . Kiefer (1961) shows that the  $D$ -optimal three-point designs are  $\mathbf{p}_3^{*(1)} = (\frac{1}{3}, 0, \frac{2}{3})$  and  $\mathbf{p}_3^{*(2)} = (\frac{2}{3}, 0, \frac{1}{3})$ , but we use this simple example to illustrate the branch-and-bound procedure.

The five nodes generated by the algorithm are listed in Table 1 in the order of their consideration and

Table 1. Nodes Generated by the Algorithm for the Example Discussed in Section 2.4

Node	Ancestor node	Constraints						$b_1(\ell, \mathbf{u})^+$	$b_2(\ell, \mathbf{u})$	$D^*$	Action
		$\ell_1$	$\ell_2$	$\ell_3$	$u_1$	$u_2$	$u_3$				
1	-	0	0	0	1	1	1	.1159E+04	.1000E+01	0	Branch on $x_3$
2	1	0	0	$\frac{1}{3}$	1	1	1	.1068E+04	.1000E+01	0	Branch on $x_1$
3	2	$\frac{1}{3}$	0	$\frac{1}{3}$	1	1	1	.8894E+00	.1000E+01	0	$D^*$ replaced by .8894E+00
4	2	0	0	$\frac{1}{3}$	0	1	1	.6695E+02	.2500E+00	.8894E+00	Terminate this branch
5	1	0	0	0	1	1	0	.1159E+04	.2500E+00	.8894E+00	Terminate this branch

<sup>+</sup>  $\alpha$  set at 0.001

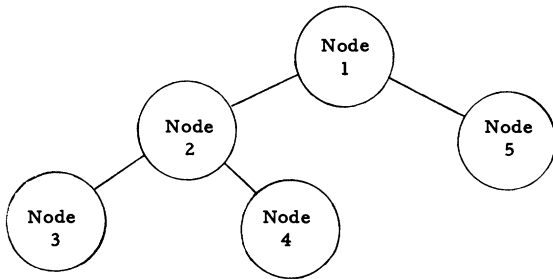


Figure 1. The Binary Tree Generated by the Algorithm for the Example Discussed in Section 2.4

Figure 1 illustrates the binary tree arrangement. We set  $\alpha$  at .001 throughout. At node 1, the root, the inequality (7) does not hold and the branching operation takes place. For this node the heuristic (8) yields a tie:  $d(x_j, l, \alpha)$  achieves a maximum at  $x_1$  and  $x_3$  but we arbitrarily choose  $x_3$  for branching. Hence, node 2 with  $l_3 = 1/3$  is constrained to have at least one design point  $x_3$ , whereas the other descendant, node 5, with  $u_3 = 0$ , allows only those three-point designs with zero design points  $x_3$ . Similarly, branching takes place at node 2, but this time the candidate  $x_1$  is indicated by (8). At node 3, since  $n - n^{(a)} = 1$ , if terms involving  $\alpha$  are neglected we have equality in the bound (6) and  $b_1(l, u)$  may be used to represent the maximum determinant of all three-point designs satisfying the node constraints, with proportional error approximately .001. The designs providing the maximum for this node immediately follow as  $\bar{d}(l, \alpha)$  in (6) is achieved for both  $x_1$  and  $x_3$  and either may be added to the two points already assigned by  $l$ , yielding the designs  $\mathbf{p}_3^{*(1)}$  and  $\mathbf{p}_3^{*(2)}$  above. For nodes 4 and 5 the inequality (7) holds,  $D^*$  is never superseded again, and the algorithm demonstrates that  $\mathbf{p}_3^{*(1)}$  and  $\mathbf{p}_3^{*(2)}$  are, in fact, the  $D_3$ -optimal designs.

2.5 Exploiting Symmetries

It may be possible to exploit symmetries when branching to impose tighter constraints on one of the descendant nodes and reduce the size of the binary tree. For a given ancestor, if there exist a  $k \times k$  matrix  $A$  with  $\{\det(A)\}^2 = 1$  and a permutation  $\pi(1), \dots, \pi(r)$  of the integers  $1, \dots, r$  such that

$$f(x_j) = Af(x_{\pi(j)}), \quad j = 1, \dots, r \quad (9)$$

$$l_j = l_{\pi(j)}, \quad j = 1, \dots, r \quad (10)$$

and

$$u_j = u_{\pi(j)}, \quad j = 1, \dots, r \quad (11)$$

then we say that the ancestor is symmetric with respect to  $\pi$ . For example, node 1 in Table 1 is symmetric with respect to  $\pi(1) = 3, \pi(2) = 2, \pi(3) = 1$  and

$$A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Consider an  $n$ -point design  $\mathbf{p}_n$  satisfying the ancestor node constraints (2), (3), and (4). The design  $\mathbf{p}_n^{(\pi)} = (p_{\pi(1)}, \dots, p_{\pi(r)})$  also satisfies (2), (3), and (4) because of (10) and (11), and some simple rearrangement and the application of (9) yield the result

$$\det\{M(\mathbf{p}_n)\} = \det\{M(\mathbf{p}_n^{(\pi)})\}. \quad (12)$$

When partitioning on  $x_{j_0}$  to form the two descendants, suppose that  $\pi(j_0) \neq j_0$ . We may now decrease two elements of  $u$  rather than just one to create the second descendant:  $u_{j_0}$  and  $u_{\pi(j_0)}$  are both reduced to  $(n_0 - 1)n^{-1}$ . To see that a  $D_n$ -optimal design is still guaranteed, consider branching on  $j_0$  as before (ignoring the symmetry) and then partitioning the second descendant by further branching on  $\pi(j_0)$ . The three resulting descendants take the constraints of the ancestor with the following adjustments:

$$l_{j_0} \text{ increased} \quad (D1)$$

$$u_{j_0} \text{ decreased, } l_{\pi(j_0)} \text{ increased} \quad (D2.1)$$

$$u_{j_0} \text{ decreased, } u_{\pi(j_0)} \text{ decreased.} \quad (D2.2)$$

Because of the symmetry of increasing the lower constraint at either  $j_0$  or  $\pi(j_0)$ , it may be shown that any  $n$ -point design permitted by descendant D2.1 is only a permutation  $\mathbf{p}_n^{(\pi)}$  of a design  $\mathbf{p}_n$  in D1 with the same determinant by (12). Hence D2.1 may be ignored and the algorithm continues with the two descendant nodes D1 and D2.2. In general, we may tighten  $t$  extra constraints on the second descendant if  $t$  permutations  $\pi_1, \dots, \pi_t$  and  $t$   $k \times k$  matrices  $A_1, \dots, A_t$  exist such that  $\{\det(A_i)\}^2 = 1$  ( $i = 1, \dots, t$ ),  $\pi_i$  and  $A_i$  satisfy (9), (10), and (11) ( $i = 1, \dots, t$ ), and  $j_0, \pi_1(j_0), \dots, \pi_t(j_0)$  are distinct.

2.6 Augmenting Designs

In the design augmentation problem we wish to add a further  $n^{(1)}$  point to  $n^{(0)}$  points at which observations have already been taken, to produce a  $D$ -optimal  $(n^{(0)} + n^{(1)})$ -point design subject to inclusion of the first  $n^{(0)}$  points.

Only the specification of the root node need be modified: we now set  $I_j = n_j^{(0)}(n^{(0)} + n^{(1)})^{-1}(j = 1, \dots, r)$ , where  $n_j^{(0)}$  observations are allocated to  $x_j$  by the initial  $n^{(0)}$  points and proceed recursively as before.

2.7 Multiple Criteria

While not explicitly stated, the branch-and-bound algorithm will not only guarantee a  $D_n$ -optimal design, but will provide a complete catalog of all  $D_n$ -optimal designs if ties occur (it is computationally simple to recover any optimal designs lost due to exploitation of symmetries when the algorithm terminates). The major criticism, however, must surely be that there is over-reliance on a single criterion,  $D$  optimality.

Two further criteria that might be considered in conjunction with  $D$  optimality when assessing an  $n$ -point design  $\mathbf{p}_n$  are the normalized maximum variance of prediction

$$v_{\max}(\mathbf{p}_n) = \max_{j=1, \dots, r} d(x_j, \mathbf{p}_n)$$

and the normalized average variance of prediction

$$v_{\text{ave}}(\mathbf{p}_n) = \frac{1}{r} \sum_{j=1}^r d(x_j, \mathbf{p}_n).$$

To incorporate these criteria, or any others proposed by the experimenter, we may change our strategy to extend the catalog of  $D_n$ -optimal designs to include all those that are "nearly"  $D_n$ -optimal in some controlled way. For example, for fixed  $\gamma(0 < \gamma < 1)$  acceptable to the experimenter, we could generate all designs  $\mathbf{p}_n$  with  $\det\{\mathbf{M}(\mathbf{p}_n)\} \geq (1 - \gamma) \det\{\mathbf{M}(\mathbf{p}_n^*)\}$  by modifying the bounding rule (7) to prevent branching if  $\min\{b_1(\mathbf{l}, \mathbf{u}), b_2(\mathbf{l}, \mathbf{u})\} < (1 - \gamma)D^*$ . It should be emphasized that on termination we know exactly how far any "nearly"  $D_n$ -optimal design is from global optimality and hence the algorithm is quite distinct from others not guaranteeing a global maximum. More importantly, though, an extensive catalog of all designs acceptable by the  $D$ -optimality criterion can be produced, which may be compared using the other criteria of interest.

### 3. DESIGNS FOR THE QUADRATIC RESPONSE SURFACE MODEL WITH THREE FACTORS AT THREE LEVELS

In this section we present some previously unknown  $D_n$ -optimal designs and illustrate the extension of the algorithm to multiple criteria.

For the design region we assume the  $3^3$  factorial arrangement where each of the three factors or explanatory variables may take one of three levels, coded as 0, 1, and 2. The quadratic response surface model for three factors has 10 parameters and may be written as

$$E(Y_{(i)}) = \beta_0 + \sum_{s=1}^3 (\beta_s x_{(i)s} + \beta_{ss} x_{(i)s}^2) + \sum_{s=1}^2 \sum_{t=s+1}^3 \beta_{st} x_{(i)s} x_{(i)t}, \quad i = 1, \dots, n,$$

where  $(x_{(i)1}, x_{(i)2}, x_{(i)3})$  are the three components of  $\mathbf{x}_{(i)}$  ( $i = 1, \dots, n$ ). Designs for  $n = 10, \dots, 20$  will be considered. This choice of design region and model yields a set of design problems within the computer time limitations of the present algorithm, but still of sufficient computational complexity that the advantages are apparent.

In Table 2 the  $D_n$ -optimal designs are compared with a compromise design that seeks to improve performance as measured by the maximum and, to a

Table 2. *Design Properties for the Quadratic Response Surface Model with Three Factors at Three Levels*

n	$D_n$ -optimal designs			Compromise design <sup>+</sup>		
	det(M)	$v_{\max}$	$v_{\text{ave}}$	det(M)	$v_{\max}$	$v_{\text{ave}}$
10	1.33E-04	27.5	13.0	1.33E-04	27.5	13.0
	1.33E-04	34.4	14.0			
11	3.24E-04	16.5	12.4	3.39E-04	17.9	11.9
12	3.39E-04	17.9	11.9			
12	3.39E-04	18.0	11.8	4.33E-04	15.4	10.3
	4.33E-04	15.4	10.3			
13	4.33E-04	15.4	10.3	4.54E-04	11.2	9.9
14	4.54E-04	11.2	9.9			
15	4.20E-04	12.9	10.5	4.10E-04	11.9	10.3
16	4.10E-04	13.5	10.4			
17	4.13E-04	14.2	10.5	3.85E-04	12.6	10.5
18	4.28E-04	14.7	10.6			
19	4.54E-04	12.4	10.4	3.76E-04	13.3	10.7
20	4.63E-04	12.1	10.2			

<sup>+</sup> As  $D_n$ -optimal design if no properties appear

lesser extent, average variances of prediction for a small loss in  $D$ -optimality efficiency. The design points are given in Table 3; where other  $D_n$ -optimal designs exist they differ only by a symmetry such as relabelling of the three factors.

The Mitchell and Bayne (1976, 1978) catalog of designs based on  $D$  optimality has recently been improved by Galil and Kiefer (1980) using a modified version of Mitchell's DETMAX algorithm. Comparison of the updated catalog with the designs supplied here shows that modified DETMAX has found a  $D_n$ -optimal design in each case, though their optimality had not previously been proved. Assuming that  $D$  optimality is not the sole concern of the experimenter, however, we now look at the case  $n = 18$  in detail. The compromise design achieves minimum  $v_{\max}(\mathbf{p}_{18})$  and minimum  $v_{\text{ave}}(\mathbf{p}_{18})$  among the class of designs such that  $\det\{\mathbf{M}(\mathbf{p}_{18})\} \geq .95 \det\{\mathbf{M}(\mathbf{p}_{18}^*)\}$  and substantially reduces the value of  $v_{\max}$  achieved by the  $D_{18}$ -optimal design. The enumeration of this case required about 63 seconds of CDC 6600 computer time and 2531 nodes were generated for a problem with approximately  $1.03 \times 10^{12}$  possible designs. As  $n$  and  $r$  increase so the number of possible designs increases exponentially and, while experience suggests that the number of nodes requiring evaluation grows more slowly than  $(r^n - 1)$ , the execution time rapidly becomes prohibitive.

### 4. CONCLUSIONS

A branch-and-bound algorithm guaranteeing a catalog of all  $D_n$ -optimal designs was presented, with an extension to improve performance by other criteria for a small loss in  $D$ -optimality efficiency. In the extended mode we are in effect using the algorithm to

Table 3. Design Points for the Quadratic Response Surface Model With Three Factors at Three Levels

n	Design <sup>+</sup>	Design points
10	D, C	002 010 021 101 112 200 202 211 220 222
	D	000 002 021 101 110 200 202 211 220 222
11	D, C	000 002 020 022 110 121 200 202 211 220 222
12	D, C	000 002 011 020 022 101 110 122 200 202 220 222
	D	000 002 011 020 022 101 110 112 200 202 220 222
13	D, C	000 002 011 020 022 101 112 120 200 202 210 221 222
14	D, C	000 002 011 020 022 101 110 112 121 200 202 211 220 222
15	D	000 002 011 012 020 022 101 110 122 200 202 212 220 221 222
	C	000 000 002 011 020 022 101 110 112 121 200 202 211 220 222
16	D	000 002 011 020 022 101 110 122 200 201 202 210 212 220 221 222
	C	000 000 002 011 020 020 022 101 110 112 121 200 202 211 220 222
17	D	000 001 002 010 012 020 022 100 102 110 121 200 202 211 220 222 222
	C	000 002 011 020 020 022 022 101 110 112 121 200 202 211 220 222 222
18	D	000 002 002 011 020 022 022 100 112 120 121 200 201 202 210 220 221 222
	C	000 001 002 010 012 020 022 100 102 111 121 200 201 202 210 212 220 222
19	D, C	000 000 002 012 020 021 022 102 111 120 122 200 201 202 210 212 220 221 222
20	D, C	000 001 002 010 012 020 021 022 100 102 110 121 200 201 202 211 220 220 222 222

<sup>+</sup> D = D<sub>n</sub> - optimal, C = Compromise.

reduce the class of all possible *n*-point designs to a smaller class acceptable by the criterion of *D* optimality. Among the reduced class a design may be chosen using secondary criteria. For moderate-sized problems of computational complexity similar to those enumerated herein it appears that advantages large enough to be of practical benefit may follow, compared with previous methods. Further work would be helpful, however, either in improving the branching strategy or tightening the bounds, to enable efficient enumeration of large problems.

A FORTRAN listing is available upon request to the author.

5. ACKNOWLEDGMENTS

This work was funded by Warren Spring Laboratory, Stevenage, England. The author would like to thank, in particular, R. Derry, P. G. Christie, and G. G. Stroud for their many practical comments on the design of experiments in chemical engineering and A. C. Atkinson for advice in the preparation of this article.

[Received June 1980. Revised May 1981.]

REFERENCES

BOX, M. J., and DRAPER, N. R. (1971), "Factorial Designs, the |X'X| Criterion, and Some Related Matters," *Technometrics*, 13, 731-742.

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

FEDOROV, V. V. (1972), "Theory of Optimal Experiments," ed. and trans. 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, 301-313.

GARFINKEL, R. S., and NEMHAUSER, G. L. (1972), "Integer Programming," New York: John Wiley.

KIEFER, J. (1958), "On the Nonrandomized Optimality and Randomized Nonoptimality of Symmetrical Designs," *Ann. Math. Statist.*, 29, 675-699.

— (1961), "Optimum Experimental Designs V, With Applications to Systematic and Rotatable Designs," *Proceedings of the Fourth Berkeley Symposium on Mathematical Statist. and Probability*, 1, 381-405.

KIEFER, J., and WOLFOWITZ, J. (1959), "Optimum Designs in Regression Problems," *Ann. Math. Statist.*, 30, 271-294.

LITTLE, J. D. C., MURTY, K. G., SWEENEY, D. W., and KAREL, C. (1963), "An Algorithm for the Traveling Salesman Problem," *Operations Research*, 11, 972-989.

MITCHELL, T. J. (1974), "An Algorithm for the Construction of *D*-optimal Experimental Designs," *Technometrics*, 16, 203-210.

MITCHELL, T. J., and BAYNE, C. K. (1976), "*D*-optimal Fractions of Three-Level Factorial Designs," ORNL/CSD-19, U. S. Dept. of Commerce, National Technical Information Service, Springfield, Va.

— (1978), "*D*-optimal Fractions of Three-Level Factorial Designs," *Technometrics*, 20, 369-380.

NEUHARDT, J. B., and BRADLEY, H. E. (1971), "On the Selec-

- tion of Multifactor Experimental Arrangements With Resource Constraints," *J. Amer. Statist. Assoc.*, 66, 618-621.
- WALD, A. (1943), "On the Efficient Design of Statistical Investigations," *Ann. Math. Statist.*, 14, 134-140.
- WYNN, H. P. (1970), "The Sequential Generation of *D*-optimum Experimental Designs," *Ann. Math. Statist.*, 41, 1655-1664.
- (1972), "Results in the Theory and Construction of *D*-optimum Experimental Designs," *J. Roy. Statist. Soc., Ser. B*, 34, 133-147.
- (1977), "Optimum Designs for Finite Populations Sampling," in *Statistical Decision Theory and Related Topics, II*, ed. S. S. Gupta and D. S. Moore, New York: Academic Press.