A Spectral Bound for D-Optimality

Chun Wa Ko
55-906 River Drive South
Jersey City, New Jersey 07310, USA

Jon Lee
Department of Mathematics
University of Kentucky
Lexington, Kentucky 40506-0027, USA

Kevin Wayne
School of Operations Research and Industrial Engineering
Cornell University
Ithaca, New York 14853, USA

Abstract. We introduce a new *spectral* bound for D-Optimal design problems, based on singular values. We compare the spectral bound to a bound based on Hadamard's inequality which was introduced by Welch. In particular, we demonstrate that (*i*) in general, neither bound dominates the other, (*ii*) the spectral bound is superior in a general situation of highly replicated designs, and (*iii*) the spectral bound is superior when a very accurate bound is required in situations of singularity. In addition, we empirically demonstrate that a branch-and-bound scheme based on *both* bounds can be quite effective in finding provably D-optimal designs.

KEY WORDS AND PHRASES: D-optimal; Design; Branch and bound; Generalized variance; Singular values.

1. Introduction.

We consider the linear model

$$y_i = x_i\beta + \epsilon_i \ (i \in N),$$

where each $1 \times m$ vector x_i is a potential design point with associated response y_i, the ϵ_i are i.i.d. with $E[\epsilon_i] = 0$ and $V[\epsilon_i] = \sigma^2$, β is an $m \times 1$ vector of parameters to be estimated, and N is an n-set of indices. We assume that there are m linearly independent design points amongst the n points. We consider the situation where we are given an e-set E of N, an f-subset F of E, and an integer s with $f \le s \le e$, and we are to choose an s-set S satisfying $F \subset S \subset E$. Of course, there is really no added generality in allowing $E \ne N$, but this notation will aid our later exposition. Many criteria have been proposed to define a "best" such set S. Let $X(S)$ be the $s \times m$ *design matrix* with rows x_i, $i \in S$, and let $D(S) := X^t(S)X(S)$. Our criterion, which we seek to maximize, is the determinant of $D(S)$. That is, we wish to solve

$$P(F,E,s): \qquad \max\Big\{\det(D(S)) \ : \ F \subset S \subset E, \ |S| = s\Big\} \ .$$

Our problem is equivalent to that of minimizing the generalized variance of the the least-squares estimator of β in the linear model.

This is the so-called "$|X^tX|$" or "D-optimality" criterion, which has been studied extensively. Smith (1918) was the first to study a formal criterion for the problem. Wald (1943) proposed the D-optimality criterion which he studied in the context of hypothesis testing. The criterion was further studied and developed by Mood (1946), and then Kiefer and Wolfowitz (1959) who coined the term "D-optimality". Methods to search for D-optimal designs have been suggested by Dykstra (1971), Federov (1972), Wynn (1970, 1972), Mitchell (1974). The papers by St. John and Draper (1975) and by Cook and Nachtsheim (1980) survey the state of the art up through the late 1970's, while Atkinson and Donev (1988), Dodge et al. (1988), and Yonchev (1988) discuss more recent progress. Welch (1982) proposed the first general algorithm for finding a provably D-optimal design. His method is based on a general framework of combinatorial optimization called *branch-and-bound*. To implement the framework Welch used lower bounds obtained by an exchange method, and two upper bounds that he proposed. One is based on Hadamard's inequality, and the other is based upon an iterative method for generating an optimal "approximate design".

In Section 2, we establish the inherent difficulty of finding a D-optimal design by showing that it is as hard as the *NP-Complete* problems of computational complexity theory. In Section 3, we establish a *spectral* upper bound on $\det(D(S))$, based on the singular values of a matrix. We also discuss a variant of the *Hadamard bound* of Welch. We confine our attention to these two bounds, and we do not discuss or use Welch's other upper bound which is much more time consuming to calculate. We demonstrate that (*i*) neither bound always dominates the other, (*ii*) the spectral bound dominates the Hadamard bound in situations where highly replicated designs are sought, and (*iii*) the spectral bound dominates the Hadamard bound when very precise bounds are required in the case in which $D(F)$ is singular. In Section 4, we describe a complete branch-and-bound algorithm based on both the Hadamard bound and the spectral bound, and we provide results of some computational experiments. We document several situations where judicious use of the spectral bound can decrease memory and time requirements of a branch-and-bound method based on just the Hadamard bound.

Using ideas similar to those of this paper, Ko, Lee and Queyranne (1993) established the theoretical complexity of finding a maximum-entropy design, and demonstrated how a bound based on the eigenvalues of a matrix could be successfully used in a branch-and-bound method.

2. Computational Complexity.

The difficulty that researchers have had in finding efficient procedures to determine D-optimal designs suggests that the problem is somehow inherently difficult. *Efficient* means that the number of basic computational steps (e.g., additions, multiplications, divisions, comparisons) of the algorithm grows no faster than a polynomial in the number of bits of a parsimonious encoding of the data. Furthermore, the number of bits necessary to encode numbers that arise in the course of the algorithm should also be bounded by a polynomial in the size of the input (see Garey and Johnson (1979) for more details). In this section, we prove that an efficient algorithm for finding a D-optimal design would yield efficient algorithms for all problems in the computational complexity class NP (see Garey and Johnson). Since most researchers in the theory of computation doubt that efficient algorithms exist for all problems in NP, this suggests that an efficient algorithm does not exist for finding D-optimal designs. More optimistically, if we could find an efficient algorithm for finding D-optimal designs, we will have found a surprising solution to the most important open problem in theoretical computer science!

We require some concepts from graph theory (see Bondy and Murty (1976)). A *finite simple graph* G is defined by its finite vertex-set $V(G)$, and its edge set $E(G)$ which is a set of two-element subsets of $V(G)$. The vertex set $\{i, j, k\}$ *induces a triangle* of G if $\{i, j\}$, $\{j, k\}$ and $\{k, i\}$ are all in $E(G)$. A graph G on $3k$ vertices can be *partitioned into triangles* if there is a partition of $V(G)$ into k disjoint three-element sets such that each three-element set induces a triangle of G.

The *NP-complete* problems are the hardest problems in NP, in the sense that the existence of an efficient algorithm for any one of them implies the existence of efficient algorithms for all problems in NP (see Garey and Johnson).

Proposition 2.1. (T.J. Schaefer (1974) unpublished; see proof in Garey and Johnson pp. 68-69). The problem of determining whether a graph G on $3k$ vertices can be partitioned into triangles is NP-complete.

We will demonstrate that the problem of partitioning a graph into triangles can be reduced to the problem of finding a D-optimal experimental design. First, we must set some notation and establish some lemmata.

For a graph G, let $A(G)$ denote the *adjacency matrix* of G; that is, the rows and columns of $A(G)$ are indexed by $V(G)$, and

$$a_{ij}(G) = \begin{cases} 1, & \text{if } \{i, j\} \in E(G); \\ 0, & \text{if } \{i, j\} \notin E(G), \end{cases}$$

for all vertices i and j. For a vertex i, let $\deg(i)$ denote the number of edges containing i. Let $D(G)$ be the matrix with rows and columns indexed by $V(G)$ defined by

$$d_{ij}(G) = \begin{cases} \deg(i), & \text{if } i = j; \\ 0, & \text{if } i \neq j, \end{cases}$$

for all vertices i and j. Finally, let $B(G)$ denote the *vertex-edge incidence matrix* of G; that is, the rows of $B(G)$ are indexed by $V(G)$, the columns of $B(G)$ are indexed by $E(G)$, and

$$b_{ie}(G) = \begin{cases} 1, & \text{if } i \in e; \\ 0, & \text{if } i \notin e, \end{cases}$$

for all vertices i and edges e.

A graph G is a $|V(G)|$-*cycle* if the there is a bijection τ from $V(G)$ to $\{1, 2, 3, \ldots |V(G)|\}$, such that

$$\begin{aligned} E(G) =& \left\{ \{\tau^{-1}(i), \tau^{-1}(i+1)\} \ : 1 \leq i \leq |V(G)| \right\} \\ & \cup \left\{ \{\tau^{-1}(|V(G)|), \tau^{-1}(1)\} \right\} . \end{aligned}$$

Lemma 2.2. If G is a cycle, then

$$|\det(B(G))| = \begin{cases} 2, & \text{if } |V(G)| \text{ is odd;} \\ 0, & \text{if } |V(G)| \text{ is even.} \end{cases}$$

Proof: We order the vertices according to τ, so that $B(G)$ has the form

$$B(G) = \begin{pmatrix} 1 & 1 & 0 & & 0 & 0 & 0 \\ 0 & 1 & 1 & & 0 & 0 & 0 \\ 0 & 0 & 1 & & 0 & 0 & 0 \\ & & & \ddots & \ddots & & \\ 0 & 0 & 0 & & 1 & 1 & 0 \\ 0 & 0 & 0 & & 0 & 1 & 1 \\ 1 & 0 & 0 & & 0 & 0 & 1 \end{pmatrix} .$$

If $|V(G)|$ is even, then the sum of the odd numbered rows (which equals the vector of all ones) is equal to the sum of the even numbered rows, hence $|\det(B(G))| = 0$. If $|V(G)|$ is odd, then we can subtract all other odd-numbered rows from the last row, and add in all even-numbered rows, which has the effect of replacing the last row of $B(G)$ with $(0, 0, 0, ..., 0, 0, 2)$. The result follows since the product of the diagonal elements of the resulting upper-triangular matrix is 2. ■

A graph G is a 1-*tree* if $|E(G)| = |V(G)|$, and there exists some edge of G whose deletion renders the graph acyclic. It is easy to see that a 1-tree consists of a "spanning tree" plus an additional edge which creates a unique cycle.

Lemma 2.3. If G is a 1-tree, then

$$|\det(B(G))| = \begin{cases} 2, & \text{if the unique cycle of } G \text{ is odd;} \\ 0, & \text{if the unique cycle of } G \text{ is even.} \end{cases}$$

Proof: If G is a cycle, then the result follows by Lemma 2.2, otherwise we proceed inductively. If G is not a cycle, then it has some vertex i that is adjacent to only one other vertex, say j. The row of $B(G)$ that corresponds to vertex i has a single nonzero entry which is in the column corresponding to edge $\{i, j\}$. We can expand the determinant along the row of $B(G)$ corresponding to vertex i, and use the inductive hypothesis and Lemma 2.2 to obtain the result. ■

A graph is *connected* if there is no nontrivial partition of the vertices into two disjoint subsets with no edge having a vertex in both parts of the partition. We note that a connected graph having equal numbers of vertices and edges is a 1-tree. A *connected component* of G is a maximal connected subgraph of G.

Lemma 2.4. Suppose that G has $V(G) = E(G)$, and G has q connected components. If each connected component is a 1-tree with an odd cycle, then $|\det(B(G))| = 2^q$ otherwise $|\det(B(G))| = 0$.

Proof: We order the vertices of G grouping together those in the same connected component so that $B(G)$ is block diagonal with q diagonal blocks. The determinant is the product of the determinants of the diagonal blocks. If each connected component has its number of edges equal to its number of vertices, then each component is a 1-tree, and the result follows from Lemma 2.3. Otherwise, G must contain a component with fewer edges than vertices. Such a component must be a tree. We can successively expand the determinant of $B(G)$ along rows corresponding to edges of the tree as in the proof of Lemma 2.3. The tree has one more vertex than its number of edges, so we will be left with a row of all zeros, in which case we see that $|\det(B(G))| = 0$. ■

We are now ready to prove our complexity result.

Proposition 2.5. If there is an efficient algorithm for the D-optimal experimental design problem (even just for the problem where $f = 0$, $e = n$, $s = m$, and each potential design point is $\{0, 1\}$-valued with two coordinates equal to 1), then every problem in NP has an efficient algorithm.

Proof: Given a finite simple graph G on $m = 3k$ vertices, let τ be a bijection from $V(G)$ to $\{1, 2, ..., |V(G)|\}$. For each $e \in E(G)$ and j, $1 \leq j \leq |V(G)|$, let

$$x_{ej} = \begin{cases} 1, & \text{if } \tau^{-1}(j) \in e; \\ 0, & \text{if } \tau^{-1}(j) \notin e. \end{cases}$$

We consider the problem of finding a D-optimal design with $N = E := E(G)$, $F = \emptyset$ and $s = m := |V(G)|$. Let S denote an arbitrary subset of N having s elements. Consider the subgraph G_S of G defined by $V(G_S) = V(G)$, and $E(G_S) = S$. We have $D(S) = B^t(G_S)B(G_S)$. Lemma 2.4 implies that $|\det(B(G_S))| \leq 2^k$, since the greatest number of disjoint 1-trees in a graph on $3k$ vertices is k, where each such 1-tree is a triangle. Furthermore, Lemma 2.4 implies that if G_S contains fewer than k triangles, then $|\det(B(G))| < 2^k$. Equivalently, $\det(D(S)) \leq 4^k$, with equality if and only if G_S is the disjoint union of k triangles. Hence, if the maximum value of $\det(D(S)) = 4^k$, then G can be partitioned into triangles, and if the maximum is less than 4^k, then there is no such partition. The result then follows from Proposition 2.1. ■

3. Upper Bounds. Initially, we make the simplifying assumption that $X(F)$ has full column rank. This implies that $D(F) = X^t(F)X(F)$ is invertible, and, moreover, that it is symmetric and positive definite. Let

$L(F)$ be the Cholesky factor of $D(F)$. That is, the invertible matrix that has all entries equal to 0 above the main diagonal, satisfying $D(F) = L(F)L^t(F)$. Such a Cholesky factor always exists and is unique for a symmetric, positive-definite matrix (see Golub and Van Loan (1983), for example). Let $\phi_i(F,E)$ denote the Euclidean norm of $x_i \cdot L^{-t}(F)$, for $i \in E \setminus F$. Let τ be a bijection from $\{1,2,...,e-f\}$ to $E \setminus F$, such that $\phi_{\tau(i)}(F,E) \geq \phi_{\tau(j)}(F,E)$ whenever $i \leq j$. Let $s_i(F,E)$ denote the i^{th} greatest number among the $\min\{e-f,m\}$ singular values of $X(E \setminus F) \cdot L^{-t}(F)$, for $1 \leq i \leq \min\{e-f,m\}$, and let $s_i(F,E) := 0$ for $\min\{e-f,m\} < i \leq e-f$. (The nonzero singular values of a matrix A are precisely the square roots of the nonzero eigenvalues of A^tA; see Golub and Van Loan, for example). We define the *Hadamard bound*

$$\mathcal{H}(F,E,s) := \det(D(F)) \prod_{i=1}^{s-f} \big(1 + \phi^2_{\tau(i)}(F,E)\big),$$

and the *spectral bound*

$$\mathcal{S}(F,E,s) := \det(D(F)) \prod_{i=1}^{s-f} \big(1 + s_i^2(F,E)\big).$$

Proposition 3.1.

$$\max\{\det(D(S)) \ : \ F \subset S \subset E, \ |S| = s\} \leq \min\{\mathcal{H}(F,E,s), \mathcal{S}(F,E,s)\} \ .$$

Proof: For a real symmetric matrix B, let $\lambda_i(B)$ denote the $i^{\rm th}$ greatest eigenvalue of B. First, we note that $D(S) = D(F) + D(S \setminus F)$. Hence

$$\begin{aligned}
\det(D(S)) =& \det\big(D(F) + D(S \setminus F)\big) \\
=& \det(D(F)) \cdot \det\big(I_{s-f} + X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)\big) \\
=& \det(D(F)) \prod_{i=1}^{s-f} \lambda_i\Big(I_{s-f} + X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)\Big) \\
=& \det(D(F)) \prod_{i=1}^{s-f} \Big(\lambda_i\Big(I_{s-f}\Big) + \lambda_i\Big(X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)\Big)\Big) \\
=& \det(D(F)) \prod_{i=1}^{s-f} \Big(1 + \lambda_i\Big(X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)\Big)\Big) \\
\leq& \det(D(F)) \prod_{i=1}^{s-f} \Big(1 + \lambda_i\Big(X(E \setminus F) \cdot D^{-1}(F) \cdot X^t(E \setminus F)\Big)\Big) \\
=& \det(D(F)) \prod_{i=1}^{s-f} \Big(1 + s_i^2\Big(X(E \setminus F) \cdot L^{-t}(F)\Big)\Big) \\
=& \mathcal{S}(F,E,s) \ .
\end{aligned}$$

thus establishing the spectral bound. We note that the inequality above follows from the *interlacing property* of singular values (see Golub and Van Loan, for example).

Additionally, we note that the $i^{\rm th}$ diagonal entry of

$$I_{s-f} + X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)$$

is $1 + \phi_i^2$. Now, since

$$I_{s-f} + X(S \setminus F) \cdot D^{-1}(F) \cdot X^t(S \setminus F)$$

is symmetric and positive definite, its determinant is no more than the product of its diagonal elements, (see Golub and Van Loan, for example) thus establishing the Hadamard bound. ■

The bound based on Hadamard's inequality in Welch has a slightly different form than ours, since in our setup we incorporate upper bounds on the number of times a design point can be replicated by explicitly including multiple copies of such points in the design matrix. Since Welch allows arbitrary replication of all design points, his bound is:

$$\det(D(F))\big(1+\phi^2_{\tau(1)}(F,E)\big)^{s-f} \ .$$

As the following example indicates, neither the Hadamard bound $\mathcal{H}$ nor the spectral bound $\mathcal{S}$ always dominates the other:

Example 3.2. Nondominance. Let

$$X = \begin{pmatrix} 1 & -1 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \\ 1 & -1 \end{pmatrix} \ ,$$

and take $F = \{1,2\}$, and $E = \{1,2,3,4,5\}$. We calculate

$$D(F) = \begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix} \ , \qquad \det(D(F)) = 1 \ ,$$

$$L(F) = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \ , \qquad L^{-t}(F) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \ ,$$

$$X(E \setminus F) \cdot L^{-t}(F) = \begin{pmatrix} 1 & 2 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} \ .$$

We have

$$s^2(F,E) = (4+\sqrt{10}, 4-\sqrt{10}, 0) \approx (7.1623, 0.8377, 0) \ ,$$

and

$$\phi^2(F,E) = (5,2,1) \ .$$

For $s = 4$, the spectral bound is sharper: We have $\mathcal{S}(F,E,s) = 15$ and $\mathcal{H}(F,E,s) = 18$, while the D-optimal design has $S = \{1,2,3,5\}$, with $\det(D(S)) = 11$. On the other hand, for $s = 3$, the Hadamard bound is sharper: In this case, we have $\mathcal{S}(F,E,s) = 5+\sqrt{10} \approx 8.1623$ and $\mathcal{H}(F,E,s) = 6$, while the D-optimal design has $S = \{1,2,3\}$, with $\det(D(S)) = 6$. ■

In the next example, we investigate how the bounds perform when we allow replication and require many design points to be selected.

Example 3.3. Heavily Replicated Designs. Let $F := \{0,1,2,...,f-1\}$. Fix a set of f m-vectors $\tilde{x}_i$, $i \in F$. We may assume that $\{\tilde{x}_i \ : \ i \in F\}$ contains m linearly independent points. Let k be a positive integer, let $E = N := \{0,1,2,...,(k+1)f-1\}$, and let $x_i := \tilde{x}_{i(\text{mod } f)}$ for $i \in N$. Let $s = s_k := f + k$. Our problem, then, is to choose $f+k$ design points from $\{x_i \ : \ i \in F\}$, *allowing arbitrary nonzero replication*, so as to produce a D-optimal design. We note that for any such problem, $\det(D(S_k))$ will behave like a polynomial in k of degree m, where S_k indicates the dependence of a D-optimal set of indices on k. In this situation we have,

$$\mathcal{H}(F,E,s_k) = \det(D(F)) \cdot (1+\phi_1^2)^k \ ,$$

where $\phi_1 := \max_{i \in F}\{\|x_i \cdot L^{-t}(F)\|_2\}$. Thus, in the present situation, the Hadamard bound grows exponentially in k. On the other hand,

$$\mathcal{S}(F,E,s_k) = \det(D(F)) \prod_{i=1}^{m} \big(1 + k \cdot s_i^2\big) \ ,$$

where s_i denotes the i^{th} greatest singular value of $X(F) \cdot L^{-t}(F)$. We note that in this case, the spectral bound increases as a polynomial in k of degree m.

Therefore, the spectral bound is within a constant factor of the D-optimal value, while the Hadamard bound is not within a subexponential factor. ■

To handle the case in which $D(F)$ is singular, we perturb the problem. Let $D_\alpha(S) = D(S) + (\alpha/n)D(N)$. Let $\bar{\nu}^2$ denote the average variance of the least squares estimators of the responses $\hat{y}_i := x_i\hat{\beta}$, over all $i \in N$, where $\hat{\beta} := D^{-1}(N)X^t y$. That is,

$$\bar{\nu}^2 = \frac{\sigma^2}{n} \sum_{i \in N} x_i D^{-1}(N) x_i^t$$

Lemma 3.4. (Mitchell (1974))

$$\lim_{\alpha \to 0} \frac{\big(\det(D_\alpha(S)) - \det(D(S))\big) \big/ \det(D(S))}{\alpha} = \bar{\nu}^2 \ .$$

Thus, we can choose some small α (Mitchell suggests .005; Welch uses .001), and change our criteria to that of maximizing $\det(D_\alpha(S))$, where

$$D_\alpha(S) := D(S) + (\alpha/n)D(N) \ .$$

Then the relative error related to using $\det(D_\alpha(S))$ rather than $\det(D(S))$ will be approximately $\alpha\bar{\nu}^2$.

For the sake of precision, let $L_\alpha(F)L_\alpha^t(F)$ be the Cholesky factorization of $D_\alpha(F)$. Let $s_i(F, E, \alpha)$ denote the i^{th} greatest number among the $\min\{e-f, m\}$ singular values of $X(E \setminus F) \cdot L_\alpha^{-t}(F)$, for $1 \le i \le \min\{e-f, m\}$, and let $s_i(F, E, \alpha) := 0$, for $\min\{e-f, m\} < i \le e-f$. We define the *spectral bound* for the perturbed problem as

$$\mathcal{S}_\alpha(F, E, s) := \det(D_\alpha(F)) \prod_{i=1}^{s-f} \big(1 + s_i^2(F, E, \alpha)\big).$$

Similarly, let $\phi_i(F, E, \alpha)$ denote the Euclidean norm of $x_i \cdot L_\alpha^{-t}(F)$, for $i \in E \setminus F$. Let τ be a bijection from $\{1, 2, ..., e-f\}$ to $E \setminus F$, such that $\phi_{\tau(i)}(F, E, \alpha) \ge \phi_{\tau(j)}(F, E, \alpha)$ whenever $i \le j$. We define the *Hadamard bound* for the perturbed problem as

$$\mathcal{H}_\alpha(F, E, s) := \det(D_\alpha(F)) \prod_{i=1}^{s-f} \big(1 + \phi_{\tau(i)}^2(F, E, \alpha)\big),$$

Alternatively, we can simply let $N' = \{(\alpha/n)x_i \ : \ i \in N\}$, in which case we have

$$\mathcal{S}_\alpha(F, E, s) = \mathcal{S}(F \cup N', E \cup N', s + n)$$

and

$$\mathcal{H}_\alpha(F, E, s) = \mathcal{H}(F \cup N', E \cup N', s + n)$$

That is, we augment the design space by the n points $\sqrt{(\alpha/n)}x_i$, and force them into the solution.

In the following example, we study how the bounds may perform when we require a high degree of precision (α near 0) in the singular case.

Example 3.5. Precise Bounds in the Singular Case. Let

$$X = \begin{pmatrix} 1 & 1 \\ -1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} ,$$

and take $F = \{1\}$, $E = \{1,2,3,4\}$ and $s = 3$. We have

$$D_\alpha(F) = \begin{pmatrix} 1+3\alpha/4 & 1 \\ 1 & 1+3\alpha/4 \end{pmatrix} ,$$

which has determinant $3\alpha(8+3\alpha)/16$. We have

$$s^2(F,E,\alpha) = (\frac{4}{\alpha} \ , \ \frac{4}{8+3\alpha} \ , \ 0) \ ,$$

and

$$\phi^2(F,E,\alpha) = (\frac{8}{3\alpha} \ , \ \frac{16+12\alpha}{24\alpha+9\alpha^2} \ , \ \frac{16+12\alpha}{24\alpha+9\alpha^2}) \ .$$

It follows that

$$\mathcal{S}_\alpha(F,E,s) = \frac{9(4+\alpha)^2}{16} \to 9 \ \ (\text{as} \ \ \alpha \to 0).$$

On the other hand,

$$\mathcal{H}_\alpha(F,E,s) = 7 + \frac{8}{3\alpha} + \frac{15\alpha}{4} + \frac{9\alpha^2}{16} \to \infty \ \ (\text{as} \ \ \alpha \to 0).$$

We note that $S = \{1,2,3\}$ is D-optimal, with $\det(D(S)) = 6$. ■

Next, we will demonstrate that the behavior of the spectral bound in Example 3.5 is not an anomaly. That is, we will establish that the spectral bound always converges as α vanishes. For a matrix A, let $\rho(A)$ denote the rank of A.

Lemma 3.6. There exist constants a_i, $m - \rho(D(F)) \le i \le m$, such that

$$\det\bigl(D_\alpha(N)\bigr) = \sum_{i=m-\rho(D(F))}^{m} a_i \alpha^i \ .$$

Proof.

$$\begin{aligned}\det\bigl(D_\alpha(N)\bigr) =& \det\Bigl(\frac{\alpha}{n}D(N)\Bigr) \cdot \det\Bigl(I + \frac{n}{\alpha}X(F)\cdot D^{-1}(N)\cdot X^t(F)\Bigr) \\ =& \alpha^m \det\Bigl(\frac{1}{n}D(N)\Bigr) \cdot \prod_{i=1}^{\rho(D(F))} \Bigl(1 + \frac{1}{\alpha}\lambda_i\bigl(nX(F)\cdot D^{-1}(N)\cdot X^t(F)\bigr)\Bigr),\end{aligned}$$

with the upper limit on the index of the product being justified by noting that

$$\rho\bigl(nX(F)\cdot D^{-1}(N)\cdot X^t(F)\bigr) \le \rho\bigl(D(F)\bigr).$$

Now

$$\prod_{i=1}^{\rho(D(F))} \Bigl(1 + \frac{1}{\alpha}\lambda_i\bigl(nX(F)\cdot D^{-1}(N)\cdot X^t(F)\bigr)\Bigr) = \sum_{i=0}^{\rho(D(F))} c_i\Bigl(\frac{1}{\alpha^i}\Bigr),$$

for some c_i $\Bigl(0 \le i \le \rho\bigl(D(F)\bigr)\Bigr)$. The result follows. ■

Lemma 3.7. There exist constants b_j, $0 \le j \le m - \rho(D(F))$, such that

$$\prod_{i=1}^{s-f}\bigl(1 + s_i^2(F,E,\alpha)\bigr) \le \sum_{j=0}^{m-\rho(D(F))} b_j\Bigl(\frac{1}{\alpha^j}\Bigr).$$

Proof. For a matrix A, let $\sigma_i(A)$ denote the i^{th} greatest singular value of A. We have

$$s_i^2(F,E,\alpha) \le \sigma_1(X(E\setminus F))\cdot\sigma_1(X^t(E\setminus F))\cdot\lambda_i\Bigl(\bigl(D(F) + \frac{\alpha}{n}D(N)\bigr)^{-1}\Bigr) \ .$$

Now,

$$\begin{aligned}\lambda_i\Big(\big(D(F)+\frac{\alpha}{n}D(N)\big)^{-1}\Big) \\ &= \lambda_{m-i+1}^{-1}\Big(D(F)+\frac{\alpha}{n}D(N)\Big) \\ &\le \frac{1}{\lambda_{m-i+1}\big(D(F)\big)\ +\ \alpha\cdot\lambda_m\Big(\frac{1}{n}D(N)\Big)}\ .\end{aligned}$$

Hence,

$$s_i^2(F,E,\alpha)\le\frac{c}{\lambda_{m-i+1}\big(D(F)\big)+b\alpha},$$

for some constants c and b. Now,

$$\lambda_{m-i+1}\big(D(F)\big)\begin{cases}=0, & \text{for } 1\le i\le m-\rho(D(F));\\ >0, & \text{for } m-\rho(D(F))<i\le m.\end{cases}$$

Therefore, there exists a constant C, such that

$$\prod_{i=1}^{s-f}\big(1+s_i^2(F,E,\alpha)\big)\le C\cdot\Big(1+\frac{c}{b\alpha}\Big)^{m-\rho(D(F))}\ .$$

The result follows. ■

Combining Lemmata 3.6 and 3.7, we immediately have the following result.

Proposition 3.8.

$$\lim_{\alpha\to 0}\mathcal{S}_\alpha(F,E,s)<\infty.$$

We can glean from the proofs of Lemmata 3.6 and 3.7 a recipe for constructing an upper bound on the limit of Proposition 3.8. It would be interesting to find an efficient method for calculating the limit exactly.

4. A Branch-and-Bound Algorithm and Experiments. Branch-and-bound is a classical algorithmic framework for solving combinatorial optimization problems. The framework does not usually lead to theoretically efficient algorithms (in the sense described in Section 2), but it often leads to algorithms that, in a practical sense, are much better than enumerating all feasible solution. Indeed, for many problems, the best known-technique is a branch-and-bound method. In general terms the method proceeds as follows: Assume that we have a maximization problem. We begin by calculating a lower bound (which is usually given by some heuristically determined feasible solution). Initially, the list of *active subproblems* is just the given problem. For each active subproblem, we have calculated an upper bound (which can often be thought of as arising by solving the problem with some constraints relaxed). The *global upper-bound* is the maximum, over all active subproblems, of the calculated upper-bounds. At each major iteration, we remove a subproblem from the list of active subproblems, and create some new subproblems which, taken together, admit all possible solutions that were admitted by the subproblem that was just removed. We calculate the upper bound for each new subproblem: If the upper bound is less than the lower bound then we discard the new subproblem (in this case, we say that the subproblem is *fathomed by bounds*). After repeated subdivision, it will be the case that some subproblems can be subdivided no further (in this case, we say that the subproblem is *fathomed by feasibility*). Occasionally, we may run further heuristics to increase the lower bound. We may stop when the list of active subproblems is empty, or when the global upper-bound and the lower bound are tolerably close. The quality of the bounds strongly influences the number of subproblems that must be solved. We note that some branch-and-bound methods (e.g., for integer programming (see Nemhauser and Wolsey (1988))) have the luxury of stumbling upon feasible solutions as they calculate the upper bounds, and thus allowing additional fathoming; unfortunately, the geometry of our design problems does not enable this to occur.

For the D-optimal design problem, we subdivide the problem $P(F,E,s)$ by choosing some $i\in E\setminus F$. We create the subproblem $P(F\cup\{i\},E\setminus\{i\},s-1)$ if $s>|F|$, and we create the subproblem $P(F,E\setminus\{i\},s)$

if $s < |E|$. Of course, if $s = |E|$ (resp. $s = |F|$), we can immediately fathom by feasibility since the only feasible solution is to choose $S = E$ (resp. $S = F$).

Our goal in computational testing is *not* to attempt to prove that the spectral bound is uniformly better than the Hadamard bound; indeed, it is *not* uniformly better. It appears that a judicious use of both bounds is warranted in a branch-and-bound framework. Our goal is to document situations where the spectral bound is useful.

For the initial lower bound, we use a *greedy* approach followed by a variant of the Wynn-Mitchell exchange algorithm. We start with $S = F$. As long as $|S| < s$, we repeatedly append $i^* \in E \setminus S$ to S, if $\det(D(S \cup \{i\}))$ is maximized by $i = i^*$, over $i \in E \setminus S$. If $D(F)$ is singular, we may seek the greatest increase in $\det\big(D_\alpha(\cdot)\big)$, for some small positive α (see Section 3). Once the greedy phase terminates, we initiate an *interchange* phase. That is, we repeatedly replace S with $S \setminus \{i^*\} \cup \{j^*\}$ if $\det(D(S \setminus \{i\} \cup \{j\}))$ is maximized by $i = i^*$ and $j = j^*$ over $i \in S$ and $j \in E \setminus S$, and $\det(D(S \setminus \{i^*\} \cup \{j*\})) > \det(D(S))$.

As our main goal was to investigate the quality of the bounds (in a branch-and-bound setting), we were not concerned with developing an extremely efficient implementation. For ease of implementation, we coded the algorithm in MATLAB, which has easy-to-use matrix facilities in addition to the usual programming constructs of a high-level language. We ran our code on an HP 9000/715 workstation with 32MB of RAM (some of the problems used more memory than this, and they ran rather slowly since information needed to be swapped in and out of main memory).

Our first set of (10) problems is based on generating design matrices in which the entries are i.i.d. uniform pseudorandom deviates. All of the problems had: $n = 30$, $m = f = 10$, $s = 20$. Our results are summarized in Table 1. The column labeled "#" contains the problem identifiers; the column labeled "Hadamard" contains the number of subproblems that were not fathomed by feasibility, when just the Hadamard bound was used (we note that the numbers of these "bound calls" compare quite favorably with the number of feasible designs $\binom{n-f}{s-f} = \binom{20}{10} = 184,756$). In parentheses, the maximum number of active subproblems is reported. The column labeled "Spectral" reports the same information when the spectral bound was used instead. The column labeled "Both" reports the same information when both bounds were used.

The second set of (10) problems is just like the first set, except the entries in the design matrices are i.i.d. normal pseudorandom deviates. These results are summarized in Table 2. We note that although the uniform problems tend to be harder than the normal problems, the relative performance of the bounds is comparable.

The third and fourth sets of (10) problems are just like the first two sets, except we have: $n = 40$, $m = f = 10$, $s = 30$. We note that the number of feasible designs is $\binom{30}{20} = 30,045,015$. For these problems, all computations were carried out using both bounds, as solving them using only one of the bounding methods proved to be too computationally burdensome. The results are summarized in Tables 3 and 4.

The fifth set of (14) problems is based on the 3-factor quadratic response model:

$$y = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \beta_3 z_3 + \beta_{11} z_1^2 + \beta_{22} z_2^2 + \beta_{33} z_3^2 + \beta_{12} z_1 z_2 + \beta_{13} z_1 z_3 + \beta_{23} z_2 z_3 + \epsilon \ .$$

Following Welch, our model allows each factor to assume 3 levels which are encoded as 0,1,2. So we are lead to a design matrix with $n = 3^3 = 27$ rows and $m = 10$ columns. We have set $f = 10$ and have fixed the 10 design points with $z_1 + z_2 + z_3 \leq 2$. We solved the problems having $12 \leq s \leq 25$. The results are summarized in Table 3, where problem "qi" has $s = i$. This series of problems gives a good indication of how the bounds may perform as we vary the number of rows s to be selected. We note that both bounds tend to fathom more easily on these problems than on the ones with random design matrices; of course, these are combinatorially simpler problems having at most $\binom{n-f}{s-f} = \binom{17}{9} = 24,310$ feasible designs.

In all of our results we noticed that the spectral bound was much more likely to fathom a subproblem that just had a point fixed out, while the Hadamard bound was much more likely to fathom a subproblem that just had a point fixed in. Indeed, even when the spectral (resp. Hadamard) bound substantially outperformed the Hadamard (resp. spectral) bound overall on a complete run on a problem, the Hadamard (resp. spectral) bound was often the winner over subproblems that just had a point fixed in (resp. out). We can easily take advantage of this, by having the order in which we compute the bounds depend on whether a point was just fixed in or fixed out; then we may not have to compute the second bound if the first one fathomed.

It turns out that in many instances, the spectral bound is a clear winner early in the branching process, but for many of those subproblems even the spectral bound is not sharp enough to fathom. Still, the spectral bound tends to do alot of the fathoming early in the branching process, while the Hadamard bound does much of its fathoming later in the branching process.

Acknowledgement

We are grateful to Maurice Queyranne for introducing us to the problem.

#	Hadamard	Spectral	Both
$u1$	4163 (623)	4438 (526)	2103 (283)
$u2$	5252 (883)	4559 (574)	2557 (432)
$u3$	18565 (2961)	7162 (633)	4665 (460)
$u4$	15808 (2406)	19250 (2066)	8202 (859)
$u5$	5976 (570)	6106 (636)	2235 (233)
$u6$	16482 (2116)	12822 (1351)	8187 (788)
$u7$	12065 (1373)	8666 (967)	4570 (491)
$u8$	17538 (2167)	23358 (2737)	11360 (1067)
$u9$	8428 (928)	9117 (741)	3104 (257)
$u10$	16036 (2398)	15185 (1689)	9752 (1107)

Table 1: Uniform $(n = 30, m = f = 10, s = 20)$

#	Hadamard	Spectral	Both
$n1$	5282 (713)	8158 (863)	3122 (356)
$n2$	16748 (2428)	7500 (1003)	5650 (744)
$n3$	5126 (608)	4537 (405)	1907 (128)
$n4$	20101 (3356)	8703 (1095)	6370 (760)
$n5$	13904 (2712)	8554 (886)	6435 (720)
$n6$	10823 (1572)	10473 (972)	5902 (504)
$n7$	9718 (1198)	9364 (916)	3493 (326)
$n8$	9481 (1298)	7467 (815)	3147 (316)
$n9$	4808 (516)	1968 (163)	1311 (96)
$n10$	16985 (2279)	14315 (1519)	7240 (647)

Table 2: Normal $(n = 30, m = f = 10, s = 20)$

#	Both
$u11$	258945 (20222)
$u12$	609663 (75948)
$u13$	446192 (45556)
$u14$	280977 (25313)
$u15$	308498 (25616)
$u16$	575589 (48052)
$u17$	192331 (21477)
$u18$	312172 (36321)
$u19$	268546 (24445)
$u20$	407029 (39894)

Table 3: Uniform ($n = 40, m = f = 10, s = 30$)

#	Both
$n11$	39015 (2598)
$n12$	174608 (16178)
$n13$	208393 (20313)
$n14$	811939 (70906)
$n15$	219492 (19767)
$n16$	131066 (9250)
$n17$	414009 (35564)
$n18$	400110 (34665)
$n19$	124665 (11541)
$n20$	248843 (23941)

Table 4: Normal ($n = 40, m = f = 10, s = 30$)

#	Hadamard	Spectral	Both
$q12$	30 (1)	53 (5)	30 (1)
$q13$	158 (9)	160 (11)	117 (6)
$q14$	392 (30)	377 (28)	266 (17)
$q15$	1508 (151)	1343 (113)	972 (72)
$q16$	3562 (428)	3078 (301)	2212 (198)
$q17$	6524 (902)	5014 (531)	3895 (377)
$q18$	10435 (1574)	8661 (934)	6914 (726)
$q19$	12313 (1987)	10190 (1208)	8484 (940)
$q20$	11281 (1916)	9467 (1245)	8051 (986)
$q21$	7927 (1384)	6594 (886)	5686 (729)
$q22$	4297 (748)	3663 (486)	3154 (432)
$q23$	1747 (292)	1490 (186)	1285 (181)
$q24$	537 (78)	472 (56)	415 (56)
$q25$	121 (13)	118 (12)	108 (11)

Table 3: Quadratic Response with 3 Factors at 3 Levels ($n = 27, m = f = 10$)

References

[1] A.C. Atkinson and A.N. Donev, Algorithms, Exact Designs and Blocking in Response Surface and Mixture Designs, in *Optimal Design and Analysis of Experiments*, Y. Dodge, V.V. Federov and H.P. Wynn (editors), Elsevier Science Publishers B.V., Amsterdam (1988), pp. 61-69.

[2] J.A. Bondy and U.S.R. Murty, *Graph Theory with Applications*, North Holland, New York (1976).

[3] M.J. Box and N.R. Draper, Factorial Designs, the $|X'X|$ Criterion, and Some Related Matters, *Technometrics* **13** (1971), 731-742.

[4] R.D. Cook and C.J. Nachtsheim, A Comparison of Algorithms for Constructing Exact D-Optimal Designs, *Technometrics* **22** (1980), 315-324.

[5] Y. Dodge, V.V. Federov and H.P. Wynn, Optimal Design of Experiments: An Overview, in *Optimal Design and Analysis of Experiments*, Y. Dodge, V.V. Federov and H.P. Wynn (editors), Elsevier Science Publishers B.V., Amsterdam (1988), pp. 1-11.

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

[7] V.V. Fedorov, *Theory of Optimal Experiments*, Academic Press, New York (1972).

[8] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., San Francisco (1979).

[9] G.H. Golub and C.F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore (1983).

[10] C.W. Ko, J. Lee and M. Queyranne, An Exact Algorithm for Maximum Entropy Sampling, to appear in *Operations Research*, (1994).

[11] T.J. Mitchell, An Algorithm for the Construction of "D-Optimal" Experimental Designs, *Technometrics* **16** (1974), 203-210.

[12] T.J. Mitchell, Computer Construction of "D-Optimal" First-Order Designs, *Technometrics* **16** (1974), 211-220.

[13] G.L. Nemhauser and L.A. Wolsey (1988), "Integer and Combinatorial Optimization," Wiley, New York.

[14] R.C. St. John and N.R. Draper, D-Optimality for Regression Designs: A Review, *Technometrics* **17** (1975), 15-23.

[15] K. Smith, On the Standard Deviations of Adjusted and Interpolated Values of an Observed Polynomial Function and its Constants and the Guidance They Give Towards a Proper Choice of the Distribution of Observations, *Biometrika* **12** (1918), 1-85.

[16] A. Wald, On the Efficient Design of Statistical Investigations, *Annals of Mathematical Statistics* **14** (1943), 134-140.

[17] W.J. Welch, Branch-and-Bound Search for Experimental Designs Based on D Optimality and Other Criteria, *Technometrics* **24** (1982), 41-48.

[18] H. Yonchev, New Computer Procedures for Constructing D-Optimal Designs, in *Optimal Design and Analysis of Experiments*, Y. Dodge, V.V. Federov and H.P. Wynn (editors), Elsevier Science Publishers B.V., Amsterdam (1988), pp. 71-80.