Journal of Statistical Planning and Inference 141 (2011) 576–587

Contents lists available at ScienceDirect

Journal of Statistical Planning and Inference

journal homepage: www.elsevier.com/locate/jspi

D-optimal minimax design criterion for two-level fractional factorial designs

Michael Wilmut, Julie Zhou *

Department of Mathematics and Statistics, University of Victoria, Victoria, BC, Canada V8W 3R4

ARTICLE INFO

Article history:
Received 15 April 2010
Received in revised form
14 June 2010
Accepted 5 July 2010
Available online 15 July 2010

Keywords:
Annealing algorithm
Model misspecification
Optimal design
Requirement set
Robust design
Sequential algorithm

ABSTRACT

A D-optimal minimax design criterion is proposed to construct two-level fractional factorial designs, which can be used to estimate a linear model with main effects and some specified interactions. D-optimal minimax designs are robust against model misspecification and have small biases if the linear model contains more interaction terms. When the D-optimal minimax criterion is compared with the D-optimal design criterion, we find that the D-optimal design criterion is quite robust against model misspecification. Lower and upper bounds derived for the loss functions of optimal designs can be used to estimate the efficiencies of any design and evaluate the effectiveness of a search algorithm. Four algorithms to search for optimal designs for any run size are discussed and compared through several examples. An annealing algorithm and a sequential algorithm are particularly effective to search for optimal designs.

1. Introduction

Suppose there are p possible factors to be investigated through a screening experiment to determine which main effects and interactions have significant influence on the outcome of an industrial process. A two-level factorial design is often used, and a full factorial design allows us to examine all the main effects and interactions and has $N=2^p$ runs. However, even if p is moderate, N can be a very large number and it may be too expensive to run an experiment with a full factorial design. In this situation, a fractional factorial design with run size $n<N$ is usually considered to study all the main effects and some interactions among the p factors.

Many design criteria have been studied in the literature. Mukerjee and Wu (2006) provide an overview on the development of factorial and fractional factorial designs and various criteria to select optimal designs. One major criterion to select an optimal fractional factorial design is to use *maximum resolution* (MR) proposed by Box and Hunter (1961a, b), where a design with a higher resolution is considered to be more desirable. Later, Fries and Hunter (1980) proposed to use *minimum aberration* (MA), which gives more discrimination than the MR. An MA design has the highest possible resolution and is an MR design. Both criteria work well under the following assumptions (Mukerjee and Wu, 2006):

(1) Lower order factorial effects are more important than higher order ones.
(2) Factorial effects of the same order are equally likely to be important.

The two criteria are not based on any particular model. However, if we know that certain effects are more important than the others and we are only interested in fitting a linear model with the important effects, some model based criteria can be

* Corresponding author. Tel.: +1 250 721 7470; fax: +1 250 7218962.
E-mail address: jzhou@uvic.ca (J. Zhou).

0378-3758/$ - see front matter
doi:10.1016/j.jspi.2010.07.002

used to select optimal designs. Model based D-optimal, A-optimal and E-optimal criteria are studied, for example in Cheng (1980) including only main effects in the linear model. The definitions of D-optimal, A-optimal and E-optimal criteria can be found in Pukelsheim (1993). The D-optimal criterion is also used in Tang and Zhou (2009) for estimating linear models with main effects and some specified two-factor interactions, and theoretical results are obtained to construct such optimal designs.

For model based design criteria, we need to address the following question. If a model is misspecified, does the optimal design provide accurate information on the effects? The least squares estimator of the model parameters is biased under model misspecification, however, a good design criterion or optimal design should have small bias. This leads us to study if a design criterion (or an optimal design) is robust against model misspecification. In this paper we attempt to answer the question for the D-optimal criterion to select optimal two-level fractional factorial designs.

First we consider a D-optimal minimax design criterion which is robust against model misspecification. The minimax design criterion has been investigated extensively for constructing regression designs, for example, Huber (1981), Wiens (1992), Shi et al. (2007), and Zhou (2001, 2008). However, the minimax design criterion has not been applied to construct fractional factorial designs. In this paper we will study the D-optimal minimax design criterion to construct two-level fractional factorial designs. Then we will explore its relationship with the D-optimal design criterion and examine the robustness of the D-optimal design criterion.

The rest of the paper is organized as follows. In Section 2, we introduce the notation for factors, a requirement set, linear models and the least squares estimation method. In Section 3, the D-optimal minimax design criterion is discussed and applied to fractional factorial designs. Theoretical results are derived to construct D-optimal minimax designs. The relationship between the D-optimal criterion and the D-optimal minimax criterion is investigated. In Section 4, several numerical methods are proposed to search for optimal designs for any run size. In Section 5, examples are given for D-optimal minimax designs, which are compared with D-optimal designs. Numerical search methods are also compared using efficiency measures. The choice of the requirement set and the estimation of the error variance are addressed briefly. Concluding remarks are in Section 6. All the proofs are included in Appendix A.

2. Notation and linear models

Let the p factors be $F_1,\ldots,F_p$, where each factor takes two levels +1 and -1. Denote $\mathbf{D}$ ($N\times p$ matrix with elements ± 1) as the full factorial design for the p factors with $N=2^p$ runs, and the ith row is denoted by $\mathbf{u}_i^T$. The interaction between factors, say F_1 and F_2, is denoted by F_1F_2. The total number of interaction terms including two-factor interactions, three-factor interactions, …, and the p-factor interaction is $N-p-1$. A requirement set $\mathcal{R}_0$ consists of all the main effects and some specified interaction terms, which we are interested in. For example, the requirement set

$$\mathcal{R}_0=\{F_1,F_2,F_3,F_1F_2,F_1F_3\} \tag{1}$$

has three main effects and two interaction terms.

Denote x_{ij} as the level of factor F_j at the ith run and define $\mathbf{x}_i=(x_{i\,1},\ldots,x_{ip})^T$, $i=1,\ldots,n$. Obviously, each design point $\mathbf{x}_i$ belongs to design space $\mathcal{S}=\{\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_N\}$, or we say that each design point $\mathbf{x}_i$ is a selected row of matrix $\mathbf{D}$. The linear model to estimate the effects for a requirement set $\mathcal{R}_0$ based on n runs is given by

$$y_i=\mathbf{z}_{\mathcal{R}_0}^T(\mathbf{x}_i)\boldsymbol{\theta}_1+\varepsilon_i,\quad i=1,\ldots,n, \tag{2}$$

where y_i is the observed response at the ith run, and $\mathbf{z}_{\mathcal{R}_0}(\mathbf{x}_i)\in R^q$ includes the grand mean term and all the terms in $\mathcal{R}_0$ evaluated at $\mathbf{x}_i$. For example, for the requirement set $\mathcal{R}_0$ in (1),

$$\mathbf{z}_{\mathcal{R}_0}(\mathbf{x}_i)=(1,x_{i1},x_{i2},x_{i3},x_{i1}x_{i2},x_{i1}x_{i3})^T,$$

and $q=6$. The random errors ε_i's are assumed to be independent with constant variance σ^2. We use ξ_n to denote a design with n design points $\mathbf{x}_1,\ldots,\mathbf{x}_n$ selected from $\mathcal{S}$ without replacement in this paper. Sometimes we just write ξ for ξ_n if the number of points in ξ is clear.

The possible full model is

$$y_i=\mathbf{z}_{\mathcal{R}_0}^T(\mathbf{x}_i)\boldsymbol{\theta}_1+g(\mathbf{x}_i)+\varepsilon_i,\quad i=1,\ldots,n, \tag{3}$$

where function $g(\mathbf{x}_i)$ is a linear function of all the other terms (interactions) that are not in $\mathcal{R}_0$, i.e.,

$$g(\mathbf{x}_i)=\mathbf{v}^T(\mathbf{x}_i)\boldsymbol{\theta}_2, \tag{4}$$

with $\boldsymbol{\theta}_2\in R^{N-q}$ an unknown constant vector. For $\mathcal{R}_0$ in (1), we have $N-q=2^3-6=2$ and

$$\mathbf{v}^T(\mathbf{x}_i)=(x_{i2}x_{i3},x_{i1}x_{i2}x_{i3}).$$

Let the complement set of a requirement set $\mathcal{R}_0$ be $\overline{\mathcal{R}}_0$ containing all the effects not in $\mathcal{R}_0$. It is clear that $\overline{\mathcal{R}}_0$ has $N-q$ interaction terms, and vector $\mathbf{v}(\mathbf{x}_i)\in R^{N-q}$ contains all the terms in $\overline{\mathcal{R}}_0$ evaluated at design point $\mathbf{x}_i$. Define the following vectors and matrices to discuss the least squares estimator (LSE) for $\boldsymbol{\theta}_1$. Let

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{Z}_1 = \begin{pmatrix} \mathbf{z}_{\mathcal{R}_0}^T(\mathbf{x}_1) \\ \vdots \\ \mathbf{z}_{\mathcal{R}_0}^T(\mathbf{x}_n) \end{pmatrix}, \quad \mathbf{Z}_2 = \begin{pmatrix} \mathbf{v}^T(\mathbf{x}_1) \\ \vdots \\ \mathbf{v}^T(\mathbf{x}_n) \end{pmatrix}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix},$$

and models (2) and (3) can be expressed in matrix form, respectively, as

$$\mathbf{y} = \mathbf{Z}_1\boldsymbol{\theta}_1 + \boldsymbol{\varepsilon} \tag{5}$$

and

$$\mathbf{y} = \mathbf{Z}_1\boldsymbol{\theta}_1 + \mathbf{Z}_2\boldsymbol{\theta}_2 + \boldsymbol{\varepsilon}. \tag{6}$$

The LSE for $\boldsymbol{\theta}_1$ is given by

$$\hat{\boldsymbol{\theta}}_1 = (\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{y},$$

which is estimated from model (5). Since the true model is (6), the LSE is possibly biased. Its bias and variance are, respectively,

$$bias(\hat{\boldsymbol{\theta}}_1) = E(\hat{\boldsymbol{\theta}}_1) - \boldsymbol{\theta}_1 = (\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{Z}_2\boldsymbol{\theta}_2, \tag{7}$$

$$Var(\hat{\boldsymbol{\theta}}_1) = \sigma^2(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}. \tag{8}$$

Model based optimal design criteria are usually formed to minimize some scalar measures of the bias vector and/or variance matrix. Ideally an optimal design should have small bias and small variance.

3. D-optimal minimax design criterion

Zhou (2008) studied a D-optimal minimax criterion for regression designs, which is robust against misspecification in the regression response. The mean squared error of the LSE is used in the criterion, so the bias of the LSE is small if there is a small departure from the assumed regression response. The models studied there include commonly used polynomial regression models and second-order linear models. The design space is a collection of N points for the independent variables, but each independent variable takes many levels. Here we follow the idea in that paper to formulate a D-optimal minimax criterion for constructing two-level fractional factorial designs. First we define the meaning of *a small departure* from the assumed (fitted) model (2). Then we define a *D-optimal minimax criterion* and derive related theoretical results for fractional factorial designs.

The fitted model for a requirement set $\mathcal{R}_0$ is model (2), and a *small departure* from it can be written as

$$y_i = \mathbf{z}_{\mathcal{R}_0}^T(\mathbf{x}_i)\boldsymbol{\theta}_1 + f(\mathbf{x}_i) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{9}$$

where departure function $f(\mathbf{x}_i)$ satisfies the following two conditions:

(C1) Function $f(\mathbf{x})$ is orthogonal to $\mathbf{z}_{\mathcal{R}_0}(\mathbf{x})$ on design space $\mathcal{S} = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$, i.e.,

$$\sum_{i=1}^{N} f(\mathbf{u}_i)\mathbf{z}_{\mathcal{R}_0}(\mathbf{u}_i) = \mathbf{0}. \tag{10}$$

(C2) Function $f(\mathbf{x})$ is bounded on $\mathcal{S}$, i.e., for a given $\alpha > 0$,

$$\frac{1}{N}\sum_{i=1}^{N} f^2(\mathbf{u}_i) \le \alpha^2. \tag{11}$$

Define a class of departure functions as $\mathcal{F} = \{f | f$ satisfies conditions (C1) and (C2)$\}$. The departure function $f(\mathbf{x})$ is closely related to function $g(\mathbf{x})$ in (4), as noted in Lemma 1.

Lemma 1. *Function f satisfying conditions* (C1) *and* (C2) *has the following form*:

$$f(\mathbf{u}_i) = \mathbf{v}^T(\mathbf{u}_i)\boldsymbol{\theta}_2, \quad i = 1, \ldots, N, \tag{12}$$

where $\boldsymbol{\theta}_2 \in R^{N-q}$ *with length* $\|\boldsymbol{\theta}_2\| \le \alpha$.

The proof is given in Appendix A. Lemma 1 indicates that the class $\mathcal{F}$ is well defined and the small departure model (9) captures all possible full (true) models in (3) with small $\boldsymbol{\theta}_2$.

Consider the LSE $\hat{\boldsymbol{\theta}}_1$ for $\boldsymbol{\theta}_1$ by fitting the assumed model (2). The mean squared error is computed under the departure model (9) and denoted by MSE($\hat{\boldsymbol{\theta}}_1, f, \xi_n$), which depends on departure function f and design ξ_n. For fixed n, a *D-optimal minimax criterion* for constructing two-level fractional factorial designs is to find a design ξ_n^* to minimize $l_D(\xi_n) \doteq \max_{f \in \mathcal{F}} \det(\mathrm{MSE}(\hat{\boldsymbol{\theta}}_1, f, \xi_n))$, and ξ_n^* is called a D-optimal minimax (DOMD) design. Since ξ_n^* minimizes the maximum value of det(*MSE*), the criterion is called a minimax criterion. In general, a minimax problem is hard to solve, but the following analytical result for the maximum value of det(*MSE*) transforms the minimax design problem to a minimization problem only.

Theorem 1. *For the class $\mathcal{F}$ satisfying conditions* (C1) *and* (C2), *we have*

$$l_D(\xi_n) = \sigma^{2q} \frac{1+\dfrac{\alpha^2}{\sigma^2}(N-\lambda_{min}(\mathbf{Z}_1^T\mathbf{Z}_1))}{\det(\mathbf{Z}_1^T\mathbf{Z}_1)},$$

where $\lambda_{min}()$ denotes the smallest eigenvalue of a matrix.

The proof is given in Appendix A. Since no point in design space $\mathcal{S}$ can be selected more than once in ξ for fractional factorial designs, the result in Theorem 1 is different from that in Zhou (2008). To study the properties of the D-optimal minimax criterion and its relationship with the D-optimal criterion for fractional factorial designs, we define the following matrix and functions:

$$\mathbf{C}(\xi_n) = \mathbf{Z}_1^T\mathbf{Z}_1, \quad L(n) = \min_{\xi_n} l_D(\xi_n),$$

$$b(\xi_n) = \lambda_{min}(\mathbf{C}(\xi_n)), \quad e(n) = \max_{\xi_n} b(\xi_n),$$

$$m(\xi_n) = \det(\mathbf{C}(\xi_n)), \quad d(n) = \max_{\xi_n} m(\xi_n).$$

The D-optimal criterion is to find a design, say ξ_n^d, to maximize $m(\xi_n)$, and ξ_n^d is called a D-optimal (DOD) design. The E-optimal criterion is to find a design, say ξ_n^e, to maximize $b(\xi_n)$, and ξ_n^e is called an E-optimal (EOD) design. The result in Theorem 1 indicates that there are relationships among DOD, DOMD and EOD designs. For regression designs, DOD and DOMD designs are different as studied in Zhou (2008). Here we are also mainly interested in comparing DOD with DOMD designs.

Functions $d(n)$, $e(n)$ and $L(n)$ are all monotonic in n, and their lower and upper bounds are given in the following lemma.

Lemma 2. *For $q \le n \le N$, we have*:

(i) $0 \le e(n-1) \le e(n) \le n$;
(ii) $0 \le d(n-1) \le d(n) \le n^q$;
(iii) $\sigma^{2q} n^{-q}(1+(\alpha^2/\sigma^2)(N-n)) \le L(n) \le L(n-1)$.

The proof is in Appendix A. The bounds in Lemma 2 are useful if we use numerical methods to search for DOMD, DOD and EOD designs.

Remarks.

R1: For a given n, if there exists a design that maximizes both functions $m(\xi_n)$ and $b(\xi_n)$, then the DOMD and DOD designs are the same for run size n.

R2: A design ξ_n is defined to be orthogonal if it satisfies $\mathbf{C}(\xi_n) = n\mathbf{I}_q$. As a special case, both $m(\xi_n)$ and $b(\xi_n)$ reach their upper bounds at orthogonal designs according to Lemma 2. Therefore orthogonal designs are DOMD, DOD and EOD designs. However, orthogonal designs exist only for a very few values of n for a given requirement set $\mathcal{R}_0$. In particular, if n is not a multiple of 4, orthogonal designs do not exist for any $\mathcal{R}_0$ with $q > 2$. If n is a multiple of 4, orthogonal designs may exist for some $\mathcal{R}_0$ (Deng and Tang, 1999).

R3: It is clear that $\mathbf{C}(\xi_N) = N\mathbf{I}_q$, thus $b(\xi_N) = N$, $m(\xi_N) = N^q$ and $l_D(\xi_N) = \sigma^{2q}N^{-q}$. The upper bounds for $d(n)$ and $e(n)$ and lower bound for $L(n)$ can be reached in Lemma 2 for n=N. However, for $n < N$, these bounds can be improved by the following results for n=$N-1$:

$$b(\xi_{N-1}) = N-q, \quad m(\xi_{N-1}) = (N-q)N^{q-1} \tag{13}$$

for any $\mathcal{R}_0$ with $q \le N-1$ and any ξ_{N-1} (see a proof in Appendix A). Thus the upper bounds for $d(n)$ and $e(n)$ can be improved as follows, for $n \le N-1$,

$$e(n) \le \min\{n, N-q\},$$

$$d(n) \le \min\left\{n^q, \left(\frac{nq-N+q}{q-1}\right)^{q-1}(N-q)\right\},$$

and the lower bound for $L(n)$ can be modified accordingly.

R4: Define a ratio parameter $v = \alpha^2/\sigma^2$. To compute DOMD designs, we only need to know the value of the ratio v, not the exact values of α^2 and σ^2. Parameter v can be also viewed as a number to measure the importance of the bias relative to the variance of the LSE in the minimax criterion. Here are two special cases: (i) when $v=0$, the minimax criterion becomes the D-optimal criterion where the variance is important and the bias is ignored; (ii) when $v \to \infty$, the minimax criterion puts more weight on the bias than the variance.

R5: In many situations, DOD and DOMD designs are different. Several examples are given in Section 5.

4. Numerical methods to compute optimal designs

In general, it is difficult to construct DOD and DOMD designs theoretically. It is also challenging to compute and find DOD and DOMD designs numerically, even for moderate p. Four numerical methods are considered here: (a) a complete search algorithm, denoted by ALC(N,n); (b) a random search algorithm, denoted by ALR(N,n,M); (c) an annealing search algorithm, denoted by ALA(N,n,T_0,a_0,N_T, M_0); and (d) a sequential search algorithm, denoted by ALQ(N,n,n_0). In all the algorithms, parameters N and n are fixed for a given problem, and the other parameters M,T_0,a_0,N_T,M_0,n_0 need to be selected or adjusted carefully to increase the efficiency and effectiveness of the algorithms. The details for the four algorithms are given below.

Algorithm ALC(N,n) searches through all possible $\binom{N}{n}$ designs to find DOD and DOMD designs. It is computational expensive to find optimal designs using the complete search, since $\binom{N}{n}$ can be very large for moderate and large value of p. The other three algorithms are developed to reduce the amount of computation and search for DOD and DOMD designs, but they are not guaranteed to find DOD or DOMD designs. The resulting designs are called near DOD or DOMD designs. In the following we use the loss function $\mathcal{L}(\xi)=(\mathcal{L}_D(\xi_n))^{1/q}/\sigma^2$ to explain how ALR(N,n,M), ALA(N,n,T_0,a_0,N_T, M_0) and ALQ(N,n,n_0) find near DOMD designs. To find near DOD designs, just use $\mathcal{L}(\xi)=-(m(\xi_n))^{1/q}$.

Algorithm ALR(N,n,M) selects M designs randomly with size n from N runs in the full factorial design **D**, computes the corresponding loss function $\mathcal{L}(\xi)$ for each design, and determines the design that minimizes $\mathcal{L}(\xi)$ to get a near DOMD design. The near DOMD and DOD designs from ALR(N,n,M) are denoted by ξ_n^{R*} and ξ_n^{Rd}, respectively. It is obvious that the effectiveness of this algorithm increases as parameter value M increases.

Algorithm ALA(N,n,T_0,a_0,N_T, M_0) is an annealing algorithm. Various annealing algorithms have been developed in the literature to compute D-optimal and minimax robust designs, for example, Fang and Wiens (2000), Haines (1987), Meyer and Nachtsheim (1988) and Zhou (2001, 2008), and they are effective to find these designs. Here we modified the annealing algorithm in Fang and Wiens (2000) to develop algorithm ALA(N,n,T_0,a_0,N_T, M_0), where parameter T_0 is the initial temperature, a_0 is the maximum number of points that are allowed to change in a design to generate a new design, N_T is the number of designs searched at each temperature, and M_0 is the total number of temperature changes. Therefore the total number of designs searched in this algorithm is $N_T \times M_0$. Here are the steps.

Step 1: Randomly select an initial design, denoted by ξ^0, with n design points from design space $\mathcal{S}$ without replacement. Let J be the number of temperature changes in the algorithm and set J=1 at the beginning.

Step 2: Compute the loss function $\mathcal{L}(\xi^0)$ and define a subset $\mathcal{S}_1$ including all the points in $\mathcal{S}$ that are not in ξ^0.

Step 3: Randomly select a number a_1 from set {1, 2,…,a_0}. Select a_1 design points from ξ^0 randomly and replace them by a_1 design points selected randomly from $\mathcal{S}_1$ to form a new design called ξ^1.

Step 4: Compute the loss function $\mathcal{L}(\xi^1)$. If $\mathcal{L}(\xi^1)<\mathcal{L}(\xi^0)$, then accept the new design. If $\mathcal{L}(\xi^1)\geq\mathcal{L}(\xi^0)$, then accept the new design with probability $p_0=\exp(-(\mathcal{L}(\xi^1)-\mathcal{L}(\xi^0))/T_0)$. If the new design is accepted, then let $\xi^0=\xi^1$. Repeat Steps 2–4 N_T times and then go to Step 5.

Step 5: Reduce the temperature by a factor of 0.9, i.e. T_0=0.9T_0, and set J=J+1. If $J \leq M_0$, then go to Step 2. Otherwise go to Step 6.

Step 6: The near DOMD design is ξ^0.

For each design problem, parameters T_0, N_T and M_0 need to be selected carefully. In general, T_0 should be large to accept a large fraction of new designs at the beginning of the search, and M_0 should be large enough to make $0.9^{M_0}T_0$ very small so that the loss function is nonincreasing at the end of the search. Parameter N_T should be large as well so that we have searched many designs at each temperature. In order to compare this algorithm with ALR(N,n,M), it is reasonable to have $M=N_T \times M_0$ so that the number of designs searched is the same for the two algorithms. Parameter a_0 can be any value less than or equal to n. Setting a_0=5 works well in the examples in Section 5. In some annealing algorithms a_0 is set to be 1. Often we construct a plot of the loss function of accepted designs versus the number of accepted designs to see if the temperature T_0 is set properly. It is also helpful to run the algorithm several times with different initial designs to search for DOMD designs. The near DOMD and DOD designs from this algorithm are denoted by ξ_n^{A*} and ξ_n^{Ad}, respectively.

Algorithm ALQ(N,n,n_0) uses a DOMD design for n=n_0, and it searches for DOMD designs for $n=n_0+1,n_0+2,\ldots$, sequentially. The resulting designs are denoted by ξ_n^{S*}. Suppose a DOMD design for n=n_0 is $\xi_{n_0}^*$, and ALQ(N,n,n_0) constructs $\xi_{n_0+1}^{S*},\xi_{n_0+2}^{S*},\ldots$, as follows. Design ξ_{i+1}^{S*} is constructed by adding a design point to design ξ_i^{S*} so that $\mathcal{L}(\xi_{i+1}^{S*})$ has the minimum value. The near DOD designs from this algorithm are denoted by ξ_n^{Sd}.

To evaluate the search algorithms, we use two efficiency measures for a design ξ_n defined by, assuming $d(n)>0$,

$$DE(\xi_n)=\left(\frac{m(\xi_n)}{d(n)}\right)^{(1/q)},\quad LE(\xi_n)=\left(\frac{L(n)}{l_D(\xi_n)}\right)^{(1/q)}. \tag{14}$$

It is obvious that $0\le DE(\xi_n),LE(\xi_n)\le 1$ for $n\ge q$. If a design has $DE(\xi_n)=1$ (or $LE(\xi_n)=1$), then it is a DOD (or DOMD) design. If a design has $DE(\xi_n)$ (or $LE(\xi_n)$) close to 1, it is D-efficient (or L-efficient) and called a near DOD (or DOMD) design. If an algorithm can produce optimal or efficient designs, it is effective and efficient.

5. Examples and discussions

Four representative examples are given in this section to compare DOMD designs with DOD designs and to compare the numerical methods. To present the results in this section, we use a standard list for the rows and columns of design **D** as follows. The levels of factor F_i are listed in column i of **D**, and they are alternated between a block of -1 (2^{i-1} times) and a block of +1 (2^{i-1} times) in N rows. Table 1 gives an example for **D** for p=3.

In Example 1, p=4, N=16, and $\binom{N}{n}$ is not large, so the complete search algorithm ALC(N,n) is applied to compute DOMD and DOD designs. In Example 2, p=5, N=32, and $\binom{N}{n}$ is quite large. For instance, with N=32 and n=12, $\binom{N}{n}=225{,}792{,}840$. However, the DOMD and DOD designs are still computed by the complete search algorithm ALC(N,n) for several values of n. Thus we can use the results to compute the efficiencies for near DOMD and DOD designs obtained by the other three algorithms, and the results are presented in Example 3. In Example 4, p=8 and N=256, so it is not feasible to perform a complete search to find DOMD and DOD designs. The three algorithms ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) are applied and compared to search for optimal designs for n=17, 18, 19 and 20.

Example 1. Consider the requirement set $\mathcal{R}_0=\{F_1,F_2,F_3,F_4,F_1F_2,F_1F_3\}$ with p=4 factors and q=7 in the linear model (2). The DOD and DOMD designs are computed for run size n=8, 9, …, 16 in this example. Since $N=2^4=16$, the number $\binom{N}{n}$ is not large and the complete search algorithm ALC(N,n) is used. From the computation, we can summarize the results as follows: (i) the DOD and DOMD designs are not unique; (ii) the set of DOD designs and the set of DOMD designs are the same; (iii) the DOMD designs are the same for all $v>0$; (iv) the DOD and DOMD designs can be constructed sequentially for run size n=8, 9,…,16. One solution of sequential designs is presented in Table 2. Fig. 1 shows the plots of: (a) $d(n)^{(1/q)}$ (determinant) versus run size n, (b) $e(n)$ (eigenvalue) versus run size n, and (c) $L(n)^{(1/q)}/\sigma^2$ (loss function) versus run size n. The ratio $v=1$. The dashed lines are the theoretical upper and lower bounds from Lemma 2 and Remark R3, and the circles are the actual maximums and minimums. It is clear that the theoretical upper bounds (lower bounds) are very close to the actual maximums (minimums), and they are reached by the actual maximums (minimums) for n=8, 15 and 16 in this example.

Table 1
A standard list for the rows and columns of design **D** for p=3.

Row	F_1 (column 1)	F_2 (column 2)	F_3 (column 3)
1	−1	−1	−1
2	+1	−1	−1
3	−1	+1	−1
4	+1	+1	−1
5	−1	−1	+1
6	+1	−1	+1
7	−1	+1	+1
8	+1	+1	+1

Table 2
DOD and DOMD designs for Example 1.

Run size n	Rows from **D** in DOD and DOMD designs
8	1,2,7,8,11,12,13,14
9	1,2,7,8,11,12,13,14,15
10	1,2,7,8,11,12,13,14,15,16
11	1,2,7,8,11,12,13,14,15,16,10
12	1,2,7,8,11,12,13,14,15,16,10,5
13	1,2,7,8,11,12,13,14,15,16,10,5,6
14	1,2,7,8,11,12,13,14,15,16,10,5,6,4
15	1,2,7,8,11,12,13,14,15,16,10,5,6,4,3
16	1,2,7,8,11,12,13,14,15,16,10,5,6,4,3,9

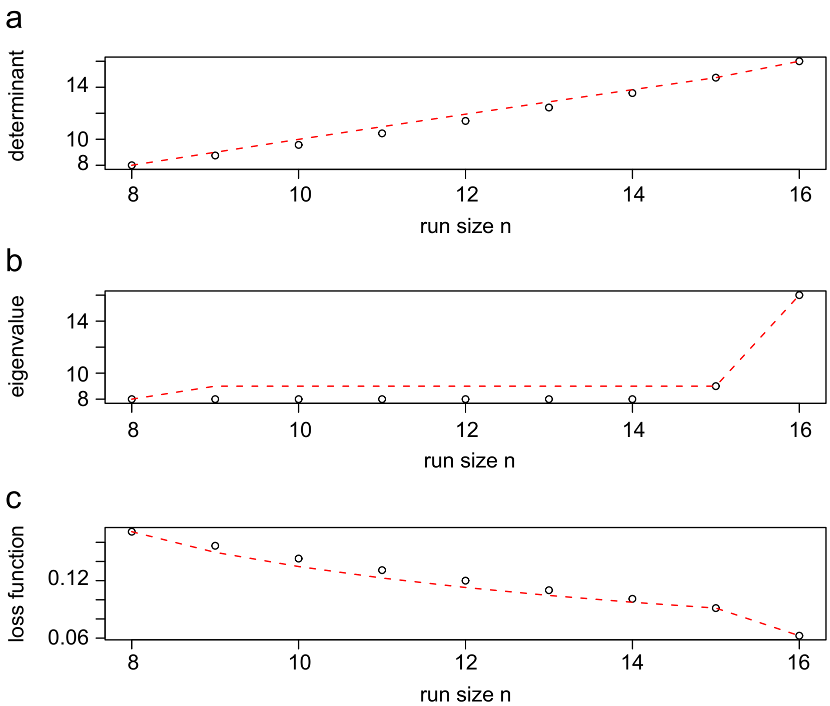

Fig. 1. The upper and lower bounds plot for Example 1: (a) $d(n)^{(1/q)}$ (determinant) versus run size n, (b) $e(n)$ (eigenvalue) versus run size n, and (c) $L(n)^{(1/q)}/\sigma^2$ (loss function) versus run size n. The dashed lines are the theoretical upper bounds in (a) and (b) and lower bound in (c) from Lemma 2 and Remark R3, and the circles are the actual maximums and minimums.

Table 3
DOMD designs (ξ^*) and DOD designs (ξ^d) for Example 2.

Run size n	Rows from **D** in optimal designs	$b(\xi_n)$	$m(\xi_n)^{(1/q)}$	$l_D(\xi_n)^{(1/q)}/\sigma^2$
8	$\xi^* = \xi^d$: 4,6,11,13,17,23,26,32	8	8	0.44100
12	$\xi^* = \xi^d$: 1,2,7,12,14,16,20,22,24,26,27,29	8	11.48151	0.30727
15	ξ^* : 1,2,3,4,7,13,14,16,21,22,24,26,27,28,31	8.70849	14.64321	0.24003
	ξ^d : 4,5,6,8,9,10,11,15,17,18,23,28,29,30,32	8	14.67206	0.240457
	ξ^e : 1,3,7,8,12,13,14,18,20,21,24,25,26,27,31	9.527864	14.48481	0.24157
16	$\xi^* = \xi^d$: 2,3,5,8,9,12,14,15,17,20,22,23,26,27,29,32	16	16	0.20960
19	ξ^* : 2,3,5,7,8,9,12,14,15,16,18,19,20,21,24,25,28,30,31	16	18.62748	0.18003
	ξ^d : 1,3,5,6,9,10,12,15,16,18,19,20,21,24,25,27,29,30,31	14.53590	18.66362	0.18166
20	$\xi^* = \xi^d$: 1,2,3,5,7,9,12,14,15,16,19,20,21,22,24,25,26,27,29,31	16	19.69617	0.17026

For n=15, design ξ^e is an EOD design.

Example 2. Consider the requirement set $\mathcal{R}_0 = \{F_1,F_2,F_3,F_4,F_5,F_1F_2,F_1F_3\}$ with p=5 factors and q=8 in model (2). The DOD and DOMD designs are computed for run size n=8, 12, 15, 16, 19 and 20 and $v = 1000$ in this example. Since N=2^5=32, the number $\binom{N}{n}$ is large. However, the results are still obtained by the complete search algorithm ALC(N,n). Table 3 lists some DOD and DOMD designs and their values of $b(\xi_n)$, $m(\xi_n)^{(1/q)}$, $l_D(\xi_n)^{(1/q)}/\sigma^2$. Again the DOD and DOMD designs are not unique. Furthermore, from the computation of this example we observe the following results:

(i) For n=8, 12, 16 and 20, the DOD and DOMD designs are the same. For n=8 and 16, the DOD and DOMD designs are orthogonal designs.
(ii) If different designs are obtained by maximizing $b(\xi_n)$ and $m(\xi_n)$, then the DOD and DOMD designs are not the same, such as in the cases of n=15 and 19.
(iii) Often DOMD designs are EOD designs, especially for large v. However, they can be different too. For example, for n=15, three different designs are obtained by maximizing $b(\xi_n)$, maximizing $m(\xi_n)$ and minimizing $l_D(\xi_n)$, respectively.

Table 4
Efficiencies for algorithms ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) in Example 3.

Efficiency	n=12	n=15	n=16	n=19	n=20
$DE(\xi_n^{Rd})$	0.988	1.0	0.986	1.0	0.996
$DE(\xi_n^{Ad})$	1.0	1.0	1.0	1.0	1.0
$DE(\xi_n^{Sd})$	0.985	1.0	1.0	0.998	0.995
$LE(\xi_n^{R*})$	0.985	1.0	0.951	1.0	0.995
$LE(\xi_n^{A*})$	1.0	1.0	1.0	1.0	1.0
$LE(\xi_n^{S*})$	0.985	0.998	1.0	1.0	0.995

The number $\binom{N}{n}$ can be very large for moderate and large p, thus it is computational expensive to find optimal designs using the complete search algorithm. In the next example, we use the three algorithms ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) to search for the DOMD and DOD designs in Example 2 and compare their efficiencies.

Example 3. Consider the same requirement set as in Example 2 and compare the three search algorithms with the complete search algorithm to find optimal designs. We can use the DOD and DOMD designs in Table 3 to compute the efficiencies for the three search algorithms. Parameter values are set as follows: M=200,000, N_T=2000, M_0=100, a_0=5, T_0=0.15 for DOD designs (T_0=0.01 for DOMD designs), and n_0=8. The DOD and DOMD design for n=8 in Table 3 is used in ALQ(N,n,n_0). Near DOD and DOMD designs are found for n=12, 15, 16, 19 and 20, and their D-efficiency and L-efficiency are computed and reported in Table 4. Note that the efficiencies for ALR(N,n,M) and ALA(N,n,T_0,a_0,N_T,M_0) are obtained by running the two algorithms 5 times and selecting the designs with the smallest loss function.

From Table 4, all the efficiencies are larger than 0.95, thus all three algorithms work well in this example. The annealing algorithm is the most effective one, since it always finds DOMD and DOD designs. To check for the convergence of the annealing algorithm ALA(N,n,T_0,a_0,N_T,M_0), two plots of loss functions are included in Fig. 2. Figs. 2(a) and (c) are the loss functions for n=12 for finding DOMD and DOD designs, respectively, and they are typical ones which show gradually decreasing loss functions. Figs. 2(b) and (d) show the distributions of the number of design points changed in accepted designs, and it indicates that, in most of the time, it changes one design point, and occasionally it changes more than one design point. Allowing the annealing algorithm to change more than one design point at a time greatly increases the effectiveness of the algorithm. The sequential algorithm ALQ(N,n,n_0) also found DOD and DOMD designs for n=15, 16 and 19, while the random algorithm ALR(N,n,M) found DOD and DOMD designs for n=15 and 19. Algorithm ALR(N,n,M) seems to be the least effective one among the three algorithms.

It is obvious that, for algorithm *ALR*(N,n,M), the efficiencies increase as M increases. For large N, M needs to be very large as well to achieve high efficiencies. Algorithm ALQ(N,n,n_0) is very fast, and the efficiencies are very high in Example 3. When N increases, the amount of computation in ALQ(N,n,n_0) increases very little.

Often $d(n)$, $e(n)$ and $L(n)$ are unknown for most values of n, but we can use their upper and lower bounds in Lemma 2 to estimate design efficiencies as follows, $DE(\xi_n) \geq (m(\xi_n))^{1/q}/n$ and $LE(\xi_n) \geq ((1+v(N-n))/n^q L_D(\xi_n))^{1/q}$. If the lower bound for efficiency $DE(\xi_n)$ (or $LE(\xi_n)$) is 1, then it is clear that design ξ_n is a DOD (or DOMD) design. However, if the lower bound is less than 1, the design might be a DOD (or DOMD) design since the actual efficiency is greater than or equal to the lower bound. In the next example, we compare the performances of the three search algorithms to find optimal designs for a problem with p=8 and N=256.

Example 4. Consider $\mathcal{R}_0 = \{F_1,F_2,F_3,F_4,F_5,F_6,F_7,F_8,F_1F_2,F_3F_4,F_5F_6,F_7F_8\}$ with p=8 and q=13 in model (2). Since N=2^p=256 is large, it is too expensive to use the complete search to find the DOD and DOMD designs. Instead we construct a resolution IV design ξ_{16}^* for n=16 using defining relation I=$F_1F_6F_7F_8$=$F_2F_5F_6F_8$= $F_3F_5F_7F_8$=$F_4F_5F_6F_7$ (Mukerjee and Wu, 2006). The design ξ_{16}^* is an orthogonal design with $\mathbf{C}(\xi_{16}^*) = 16\mathbf{I}$. Thus ξ_{16}^* is both a DOD and a DOMD design, which contains the following rows of **D**: 1, 31, 44, 54, 78, 84, 103, 121, 136, 154, 173, 179, 203, 213, 226, 256.

Three algorithms ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) are applied to find optimal designs for n=17, 18, 19, 20 and $v = 1000$, and parameter values are set at M=500,000, N_T=2500, M_0=200, a_0=5, T_0=0.15 for DOD designs (T_0=0.005 for DOMD designs), and n_0=16. The above ξ_{16}^* is used in ALQ(N,n,n_0). The minimum values of $-m(\xi_n)^{1/q}$ and $l_D(\xi_n)^{1/q}/\sigma^2$ and estimated efficiencies from the three algorithms are presented in Table 5. For sequential designs, we get $\xi_n^{Sd} = \xi_n^{S*}$, n=17, 18, 19, 20, and they are formed by adding the following rows, 2, 47, 71 and 97 to ξ_{16}^* sequentially. The estimated efficiencies from algorithms ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) are very similar and all above 0.950, so they are effective to find near DOD and DOMD designs. However, algorithm ALR(N,n,M) is not effective in this example, since the estimated efficiencies are not high (0.823–0.872).

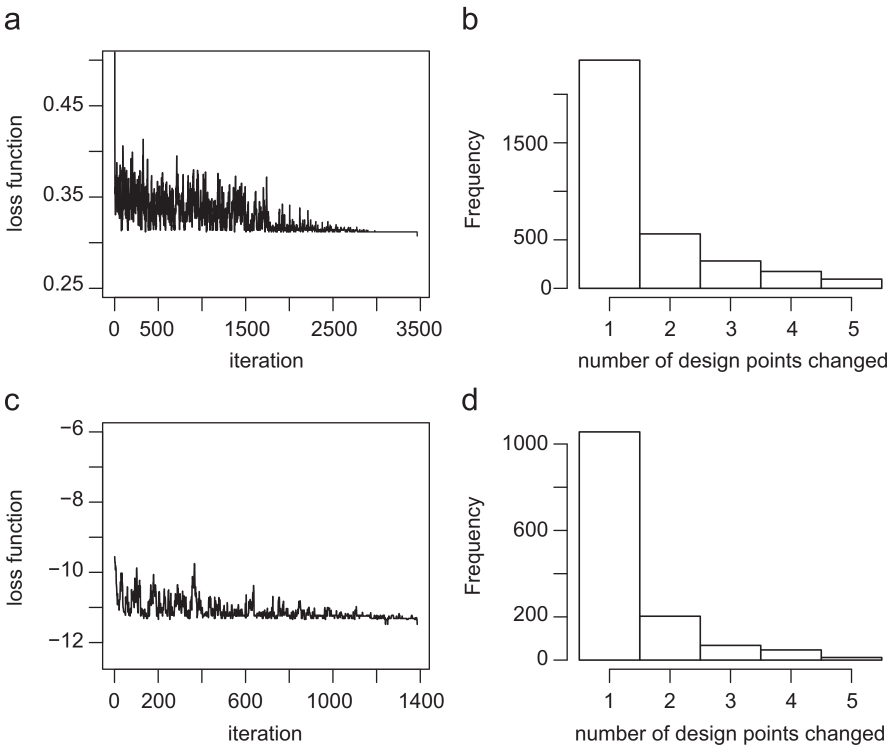

Fig. 2. The loss functions and the distributions of the number of design points changed in accepted designs for the annealing algorithm ALA(N,n,T_0,a_0,N_T,M_0) in Example 3 for n=12: (a) the loss function for finding the DOMD design, (b) the distribution of the number of design points changed in accepted designs for finding the DOMD design, (c) the loss function for finding the DOD design, (d) the distribution of the number of design points changed in accepted designs for finding the DOD design.

Table 5
Minimum loss functions and estimated efficiencies for the near DOD and DOMD designs from the three algorithms ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) in Example 4.

	n=17	n=18	n=19	n=20
DOD designs				
$-m(\xi_n^{Rd})^{1/q}$	−14.056	−15.272	−16.382	−17.440
Estimated $DE(\xi_n^{Rd})$	0.827	0.848	0.862	0.872
$-m(\xi_n^{Ad})^{1/q}$	−16.411	−17.531	−18.349	−19.246
Estimated $DE(\xi_n^{Ad})$	0.965	0.974	0.966	0.962
$-m(\xi_n^{Sd})^{1/q}$	−16.749	−17.531	−18.349	−19.203
Estimated $DE(\xi_n^{Sd})$	0.985	0.974	0.966	0.960
DOMD designs				
$l_D(\xi_n^{R*})^{1/q}/\sigma^2$	0.185	0.170	0.158	0.149
Estimated $LE(\xi_n^{R*})$	0.823	0.845	0.863	0.868
$l_D(\xi_n^{A*})^{1/q}/\sigma^2$	0.158	0.148	0.141	0.135
Estimated $LE(\xi_n^{A*})$	0.963	0.973	0.965	0.959
$l_D(\xi_n^{S*})^{1/q}/\sigma^2$	0.155	0.148	0.141	0.135
Estimated $LE(\xi_n^{S*})$	0.984	0.973	0.967	0.959

Often one more search is performed to get better near optimal designs starting with some near DOD (or DOMD) designs. We can use algorithm ALA(N,n,T_0,a_0,N_T,M_0) with T_0=0, so the loss functions are not allowed to increase in the search. For instance, in Example 4, we can take ξ_{20}^{S*} as an initial design, use a_0=5, N_T=250,000 and M_0=2 to run ALA(N,n,T_0,a_0,N_T,M_0), and get a design ξ_{20} with the following rows: 1, 31, 44, 54, 78, 84, 87, 102, 107, 121, 136, 154, 173, 179, 203, 213, 218, 226, 231, 256. This design has $-m(\xi_{20})^{1/q}=-19.293$ and $l_D(\xi_{20})^{1/q}/\sigma^2=0.134$, which are smaller than the corresponding ones for n=20 in Table 5.

Our study shows that DOD designs are DOMD designs for some cases. When DOD and DOMD designs are different, DOMD designs are D-efficient with high efficiency and DOD designs are L-efficient with high efficiency. For example, consider optimal designs for n=15 in Example 2. The D-efficiency for the DOMD design is $DE(\xi_n^*) = 14.64321/14.67206 = 0.998$, while the L-efficiency for the DOD design is $LE(\xi_n^d) = 0.24003/0.240457 = 0.998$. Thus DOD and DOMD designs are the same or very similar for all the examples we studied. This suggests that DOD designs also have small biases, and the D-optimal design criterion is robust against model misspecification.

Both the D-optimal and D-optimal minimax design criteria are model based. A requirement set $\mathcal{R}_0$ of the possible important effects needs to be specified before finding the DOD and DOMD designs. Here is a procedure to apply the DOD and DOMD designs in practice.

(P1) Specify a requirement set $\mathcal{R}_0$ that includes the important effects to be investigated. Suppose that there are $q-1$ effects in $\mathcal{R}_0$.
(P2) Determine the run size n based on the resource available where n is at least q.
(P3) Write out the linear model (5) for $\mathcal{R}_0$.
(P4) Use the loss functions $l_D(\xi_n)$ in Theorem 1 and $-m(\xi_n)$ to construct the DOMD and DOD designs, respectively.
(P5) Perform the experiment using the DOMD or the DOD design.
(P6) Fit model (5) and make inferences for the effects in $\mathcal{R}_0$.

When $n > q$, we still hope that all the effects in $\overline{\mathcal{R}}_0$ are zero or negligible, and the error variance σ^2 can be estimated using $n-q$ degrees of freedom. However, if some of the effects in $\overline{\mathcal{R}}_0$ are significant, the DOMD designs provide assurance that the mean squared error of the estimates of the effects in $\mathcal{R}_0$ is small. In this situation, the error variance tends to be overestimated by using $n-q$ degrees of freedom, but the DOMD designs may allow us to fit a larger model than model (5) and make inferences.

If $n > q$, we may be able to use a larger requirement set, say $\mathcal{R}_1$ ($\supset \mathcal{R}_0$) with $n-1$ effects, to construct the DOD or DOMD designs. If no information is available about which effects should be added to form $\mathcal{R}_1$, we recommend adding lower order factorial effects, starting with two-factor interactions, then three-factor interactions, and so on. This will tend to make the DOD and DOMD design criteria more consistent with the maximum generalized resolution and minimum aberration criteria. However, more work is needed to obtain theoretical results on this issue. When the DOD and DOMD designs are different or there are many DOMD designs, a secondary criterion, such as the minimum aberration, may be used to choose the one with the smallest aberration. The minimum aberration criterion has been studied extensively when n is a multiple of 4, but more study is needed to investigate this criterion when n is not a multiple of 4.

6. Conclusions

In this paper the D-optimal minimax design criterion is proposed to construct two-level fractional factorial designs. The resulting DOMD designs are robust against model misspecification. Furthermore, DOMD and DOD designs are compared to investigate the robustness of the D-optimal design criterion, and our study shows that the D-optimal criterion is robust. Theoretical results are obtained for the lower bounds of the loss functions for any run size n, which are useful when estimating efficiencies for any design and finding DOD and DOMD designs.

To construct DOD and DOMD designs, four numerical algorithms are considered and compared. For small p, it is feasible to run the complete search algorithm to obtain DOD and DOMD designs. However, for moderate and large p it is not feasible to run the complete search algorithm, so the other three algorithms, ALR(N,n,M), ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0), can be applied to produce near DOD and DOMD designs. Design efficiencies can be computed or estimated to determine if a design is a DOD (DOMD) design or a near DOD (DOMD) design. Furthermore, the estimated design efficiencies can be used to see if a design is close to an optimal design and evaluate the effectiveness of a numerical algorithm. Algorithms ALA(N,n,T_0,a_0,N_T,M_0) and ALQ(N,n,n_0) produce consistent results for all the examples and are more effective and efficient than ALR(N,n,M). In practice, we can run the annealing algorithm one more time with T_0=0 and starting with a near DOD or DOMD design to search for DOD and DOMD designs. Allowing the annealing algorithm to change more than one design point at a time greatly increases the algorithm's efficiency.

In general, it is difficult to construct DOD and DOMD designs analytically. However, it may be possible to construct those designs for some models and run sizes, which is a future research topic. As a special case, if orthogonal designs exist, then they are DOD and DOMD designs and these can be constructed analytically as in Example 4 for n=16. Also, it will be interesting to study the properties of the near DOD or DOMD designs from the sequential algorithm ALQ(N,n,n_0) and establish some theoretical results. When p increases, the amount of computation increases very little for ALQ(N,n,n_0) which makes it attractive. The examples in Section 5 show that ALQ(N,n,n_0) is very effective. In addition, sequential designs are especially useful in practice when we have the option to add more runs to an existing optimal or near optimal design during the running of the experiment.

Acknowledgement

This research work is supported by Discovery Grants from the Natural Science and Engineering Research Council of Canada.

Appendix A. Proofs

Define the following matrices to prove Lemma 1 and Theorem 1. Matrix $\mathbf{W}_1$ is the model matrix for $\mathcal{R}_0$ on $\mathcal{S}$, and matrix $\mathbf{W}_2$ is the model matrix for $\overline{\mathcal{R}}_0$ (without the grand mean term), i.e.,

$$\mathbf{W}_1 = \begin{pmatrix} \mathbf{z}_{\mathcal{R}_0}^T(\mathbf{u}_1) \\ \vdots \\ \mathbf{z}_{\mathcal{R}_0}^T(\mathbf{u}_N) \end{pmatrix}, \quad \mathbf{W}_2 = \begin{pmatrix} \mathbf{v}^T(\mathbf{u}_1) \\ \vdots \\ \mathbf{v}^T(\mathbf{u}_N) \end{pmatrix}.$$

Notice that

$$(\mathbf{W}_1,\mathbf{W}_2)^T(\mathbf{W}_1,\mathbf{W}_2) = N\mathbf{I}_N, \tag{15}$$

$$(\mathbf{W}_1,\mathbf{W}_2)(\mathbf{W}_1,\mathbf{W}_2)^T = N\mathbf{I}_N, \tag{16}$$

where $\mathbf{I}_N$ is the $N\times N$ identity matrix.

Proof of Lemma 1. Define vector $\mathbf{f}_N$=$(f(\mathbf{u}_1),\ldots,f(\mathbf{u}_N))^T$, then condition (C1) can be written as $\mathbf{W}_1^T\,\mathbf{f}_N$=$\mathbf{0}$, which are linear equations in $\mathbf{f}_N$. Using linear algebra, $\mathbf{f}_N$ can be solved easily. Notice that, from (15), $\mathbf{W}_1^T\mathbf{W}_2$=$\mathbf{0}$, the rank of $\mathbf{W}_1$ is q and the rank of $\mathbf{W}_2$ is $N-q$. Thus $\mathbf{f}_N = \mathbf{W}_2\boldsymbol{\theta}_2$ with $\boldsymbol{\theta}_2 \in R^{N-q}$. Condition (C2) requires that

$$\frac{1}{N}\mathbf{f}_N^T\mathbf{f}_N = \frac{1}{N}\boldsymbol{\theta}_2^T\mathbf{W}_2^T\mathbf{W}_2\boldsymbol{\theta}_2 = \frac{1}{N}\boldsymbol{\theta}_2^T N\mathbf{I}_{N-q}\boldsymbol{\theta}_2 = \boldsymbol{\theta}_2^T\boldsymbol{\theta}_2 \le \alpha^2,$$

where $\mathbf{W}_2^T\mathbf{W}_2$=$N\mathbf{I}_{N-q}$ follows from (15). Therefore $\mathbf{f}_N = \mathbf{W}_2\boldsymbol{\theta}_2$ with $\|\boldsymbol{\theta}_2\| \le \alpha$, which is the result in (12). This completes the proof. $\square$

Proof of Theorem 1. Define vector $\mathbf{f}_\xi = (f(\mathbf{x}_1),\ldots,f(\mathbf{x}_n))^T$, then, under departure model (9) and from Lemma 1,

$$\mathrm{MSE}(\hat{\boldsymbol{\theta}}_1,f,\xi)) = \sigma^2(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1} + (\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{f}_\xi\mathbf{f}_\xi^T\mathbf{Z}_1(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1} = \sigma^2(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1} + (\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{Z}_2\boldsymbol{\theta}_2\boldsymbol{\theta}_2^T\mathbf{Z}_2^T\mathbf{Z}_1(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}$$

and

$$\begin{aligned} l_D(\xi) &= \max_{f\in\mathcal{F}}\det(\mathrm{MSE}(\hat{\boldsymbol{\theta}}_1,f,\xi)) = \max_{\|\boldsymbol{\theta}_2\|\le\alpha}\sigma^{2q}\det((\mathbf{Z}_1^T\mathbf{Z}_1)^{-1})\left(1+\frac{1}{\sigma^2}\boldsymbol{\theta}_2^T\mathbf{Z}_2^T\mathbf{Z}_1(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{Z}_2\boldsymbol{\theta}_2\right) \\ &= \sigma^{2q}\det((\mathbf{Z}_1^T\mathbf{Z}_1)^{-1})\left(1+\frac{\alpha^2}{\sigma^2}\lambda_{max}(\mathbf{A})\right), \end{aligned} \tag{17}$$

where $\lambda_{max}(\mathbf{A})$ is the largest eigenvalue of matrix $\mathbf{A}$, and $\mathbf{A}$= $\mathbf{Z}_2^T\mathbf{Z}_1(\mathbf{Z}_1^T\mathbf{Z}_1)^{-1}\mathbf{Z}_1^T\mathbf{Z}_2$. A design ξ_n selected without replacement from design space $\mathcal{S}$ can be represented through a frequency vector $\mathbf{n}$ defined as $(n_1, \ldots, n_N)$, where n_i=0 or 1. If point $\mathbf{u}_i$ is in ξ, then n_i=1 otherwise 0. It is obvious that $\sum_{i=1}^N n_i = n$. Define an $N\times N$ diagonal matrix $\mathbf{M}$= $diag(n_1,\ldots, n_N)$, then it is easy to verify that $\mathbf{Z}_1^T\mathbf{Z}_1$=$\mathbf{W}_1^T\mathbf{M}\mathbf{W}_1$ and $\mathbf{Z}_1^T\mathbf{Z}_2$=$\mathbf{W}_1^T\mathbf{M}\mathbf{W}_2$. Now

$$\begin{aligned} \lambda_{max}(\mathbf{A}) &= \lambda_{max}(\mathbf{W}_2^T\mathbf{M}\mathbf{W}_1(\mathbf{W}_1^T\mathbf{M}\mathbf{W}_1)^{-1}\mathbf{W}_1^T\mathbf{M}\mathbf{W}_2) = \lambda_{max}((\mathbf{W}_1^T\mathbf{M}\mathbf{W}_1)^{-1}\mathbf{W}_1^T\mathbf{M}\mathbf{W}_2\mathbf{W}_2^T\mathbf{M}\mathbf{W}_1) \\ &= \lambda_{max}((\mathbf{W}_1^T\mathbf{M}\mathbf{W}_1)^{-1}\mathbf{W}_1^T\mathbf{M}(N\mathbf{I}-\mathbf{W}_1\mathbf{W}_1^T)\mathbf{M}\mathbf{W}_1) = \lambda_{max}(N\mathbf{I}-\mathbf{W}_1^T\mathbf{M}\mathbf{W}_1) \quad \text{using } \mathbf{M}^2=\mathbf{M} \quad \text{by (16)} \\ &= \lambda_{max}(N\mathbf{I}-\mathbf{Z}_1^T\mathbf{Z}_1) = N-\lambda_{min}(\mathbf{Z}_1^T\mathbf{Z}_1). \end{aligned}$$

Putting this result of $\lambda_{max}(\mathbf{A})$ in (17), we get the result for Theorem 1. $\square$

Proof of Lemma 2. For a given design ξ_{n-1} with $n-1$ design points, define an add-one new design $\xi_{n-1,1}$ by including all $n-1$ design points from ξ_{n-1} and one more point, say $\mathbf{u}_k$, from the rest of design space $\mathcal{S}$. The design $\xi_{n-1,1}$ has n points. Suppose design ξ_{n-1}^e maximizes $b(\xi_{n-1})$, and an add-one new design is denoted by $\xi_{n-1,1}^e$. Then $\mathbf{C}(\xi_{n-1,1}^e) = \mathbf{C}(\xi_{n-1}^e)+\mathbf{z}_{\mathcal{R}_0}(\mathbf{u}_k)\mathbf{z}_{\mathcal{R}_0}^T(\mathbf{u}_k)$, and

$$\lambda_{min}(\mathbf{C}(\xi_{n-1,1}^e)) = \lambda_{min}(\mathbf{C}(\xi_{n-1}^e)+\mathbf{z}_{\mathcal{R}_0}(\mathbf{u}_k)\mathbf{z}_{\mathcal{R}_0}^T(\mathbf{u}_k)) \ge \lambda_{min}(\mathbf{C}(\xi_{n-1}^e)).$$

Thus

$$e(n) = \max_{\xi_n} b(\xi_n) \ge b(\xi_{n-1,1}^e) = \lambda_{min}(\mathbf{C}(\xi_{n-1,1}^e)) \ge \lambda_{min}(\mathbf{C}(\xi_{n-1}^e)) = \max_{\xi_{n-1}} b(\xi_{n-1}) = e(n-1).$$

This is a part of the result of Lemma 2(i), and the other part is easy to show. Since trace$(\mathbf{C}(\xi_n)) = nq$ for any ξ_n, it is obvious that $\lambda_{min}(\mathbf{C}(\xi_n)) \le n$.

The results (ii) and (iii) can be proved similarly. Briefly for the result (ii), suppose ξ_{n-1}^d maximizes $m(\xi_{n-1})$, then an add-one design $\xi_{n-1,1}^d$ has $m(\xi_{n-1,1}^d) \ge m(\xi_{n-1}^d) = \max_{\xi_{n-1}} m(\xi_{n-1})$, which implies the result (ii). The upper bound for $m(\xi_n)$

can be shown by using the result in Bellman (1960, p. 126) and noticing that $\mathbf{C}(\xi_n)$ is a positive definite matrix with diagonal elements $n,\ldots,n$. For the result (iii), we start with the design ξ^*_{n-1} which minimizes $l_D(\xi_{n-1})$, and denote an add-one design by $\xi^*_{n-1,1}$. For design $\xi^*_{n-1,1}$, $b(\xi^*_{n-1,1}) \geq b(\xi^*_{n-1})$ and $m(\xi^*_{n-1,1}) \geq m(\xi^*_{n-1})$. Therefore $l_D(\xi^*_{n-1,1}) \leq l_D(\xi^*_{n-1}) = \min_{\xi_{n-1}} l_D(\xi_{n-1})$. The lower bound for $L(n)$ is derived by using the upper bounds for $e(n)$ and $d(n)$. $\square$

Proof of the results in (13). For a design ξ with $N-1$ points, there is only one point in $\mathcal{S}$ that is not in ξ, say point $\mathbf{u}_k$. Then

$$\mathbf{C}(\xi_{N-1}) = \mathbf{C}(\xi_N) - \mathbf{z}_{\mathcal{R}_0}(\mathbf{u}_k)\mathbf{z}^T_{\mathcal{R}_0}(\mathbf{u}_k) = N\mathbf{I}_q - \mathbf{z}_{\mathcal{R}_0}(\mathbf{u}_k)\mathbf{z}^T_{\mathcal{R}_0}(\mathbf{u}_k).$$

Now it is easy to see that, for any $\mathcal{R}_0$, matrix $\mathbf{C}(\xi_{N-1})$ has $q-1$ eigenvalues at N and one eigenvalue at $N-q$. Thus $b(\xi_{N-1}) = N-q$ and $m(\xi_{N-1}) = (N-q)N^{(q-1)}$. $\square$

References

Bellman, R., 1960. Introduction to Matrix Analysis. McGraw-Hill, New York.

Box, G.E.P., Hunter, J.S., 1961a. The 2^{k-p} fractional factorial designs. Technometrics 3, 311–352.

Box, G.E.P., Hunter, J.S., 1961b. The 2^{k-p} fractional factorial designs, II. Technometrics 3, 449–458.

Cheng, C.S., 1980. Orthogonal arrays with variable numbers of symbols. Annals of Statistics 8, 447–453.

Deng, L., Tang, B., 1999. Generalized resolution and minimum aberration criteria for Plackett–Burman and other nonregular factorial designs. Statistica Sinica 9, 1071–1082.

Fang, Z., Wiens, D.P., 2000. Integer-valued, minimax robust designs for estimation and extrapolation in heteroscedastic, approximately linear models. Journal of the American Statistical Association 95, 807–818.

Fries, A., Hunter, W.G., 1980. Minimum aberration 2^{k-p} designs. Technometrics 22, 601–608.

Haines, L.M., 1987. The application of the annealing algorithm to the construction of the exact optimal designs for linear regression models. Technometrics 29, 439–447.

Huber, P.J., 1981. Robust Statistics. Wiley, New York.

Meyer, R.K., Nachtsheim, C.J., 1988. Constructing exact D-optimal experimental designs by simulated annealing. American Journal of Mathematical and Management Sciences 3/4, 329–359.

Mukerjee, R., Wu, C.F.J., 2006. A Modern Theory of Factorial Designs. Springer, New York.

Pukelsheim, F., 1993. Optimal Design of Experiments. Wiley, New York.

Shi, P., Ye, J., Zhou, J., 2007. Discrete minimax designs for regression models with autocorrelated MA errors. Journal of Statistical Planning and Inference 137, 2721–2731.

Tang, B., Zhou, J., 2009. Existence and construction of two-level orthogonal arrays for estimating main effects and some specified two-factor interactions. Statistica Sinica 19, 1193–1201.

Wiens, D.P., 1992. Minimax designs for approximately linear regression. Journal of Statistical Planning and Inference 31, 353–371.

Zhou, J., 2001. Integer-valued, minimax robust designs for approximately linear models with correlated errors. Communications in Statistics: Theory and Methods 30, 21–39.

Zhou, J., 2008. D-optimal regression designs on discrete design space. Journal of Statistical Planning and Inference 138, 4081–4092.