On Multiple-Objective Nonlinear Optimal Designs

Qianshun Cheng, Dibyen Majumdar, and Min Yang

December 1, 2015

Abstract

Experiments with multiple objectives form a staple diet of modern scientific research. Deriving optimal designs with multiple objectives is a long-standing challenging problem with few tools available. The few existing approaches cannot provide a satisfied solution in general: either the computation is very expensive or a satisfied solution is not guaranteed. There is need for a general approach which can effectively derive multi-objective optimal designs. A novel algorithm is proposed to address this literature gap. We prove convergence of this algorithm, and show in various examples that the new algorithm can derive the true solutions with high speed.

1 Introduction

With the development of computational technology, nonlinear models have become more feasible and popular. An optimal/efficient design can improve the accuracy of statistical inferences with a given sample size or reduce the sample size needed for a pre-specified accuracy. A major complication in studying optimal designs for nonlinear models is that information matrices and thus optimal designs depend on the unknown model parameters. A common approach to solve this problem is to use locally optimal designs, which are based on the best guess of the parameters (Chernoff, H. 1953.). This strategy fits the popular multi-stages design well, an approach that has gained a lot of popularity in practice in the past decade. An initial experiment is conducted to get a better idea about the unknown parameters. The resulting initial parameter estimations are then used as the "best guess" of unknown parameters, based on which the next stage design is selected. The strategy can be carried on so on and so forth. (Hereafter, the word "locally" is omitted for simplicity.)

The selection of an optimal design depends on the goals of the experiment. For example, if the goal is to minimizes the jointed confidence ellipsoid of a parameter vector, a D-optimal design is desired. If the goal is to test hypothesises, a design with maximum power should be selected. The classical optimal design theory focus on optimizing one single objective function, such as D-optimal design, or a design with maximum power.

In practice, however, it is common for a experimenter to have multiple objectives. A typical example is multiple comparisons study, where there are multiple hypotheses testings. In a neurological stimulus-response experiment (Rosenberger and Grill, 1997), the main interests were in estimation of LD_{25} (the lethal dose that causes death for 25% of a study population), LD_{50}, and LD_{75}. An optimal design based on one specific objective could be a diaster for other objectives. In a multiple comparison study, a design with maximum power for one of the hypotheses may not have good power for the other tests. In an example with four objectives, (Clyde and Chaloner, 1996) showed that the optimal design for one of the objectives reached efficiencies of only 7%, 10%, and 39% for the others.

Although the clear importance of design for multiple objectives, little progress have been made due to the complexity of design for nonlinear models. (Yang and Stufken, 2009; Yang and Stufken, 2009; Yang, 2010; Dette, Melas, 2011; Dette, Schorning, 2013) obtained a series of unifying results so called "complete classes" of designs. They show that we can focus on a subclass of designs with a simple form, which is dominating in the Loewner sense, implying that for each design and optimality criterion there is a design in the subclass which is at least as good. These results are big steps towards simplifying design search for nonlinear models even for the multiple objectives design problems. A research gap, however, still exists. Even for the single-objective design problems, except for some special cases, it seems impossible to find the optimal design analytically, and we have to rely on a numerical solution. While we can focus on designs of a simple form, the numerical computation may still be problematic. Recent progress in developing new efficient algorithms makes it possible. (Yang, Biedermann and Tang, 2013) proposed a general and efficient algorithm (the optimal weights exchange algorithm - OWEA) which can identify an optimal design quickly regardless of optimality criteria and parameters of interest. While the new algorithm is for single objective design problem, it provides foundations for deriving the multiple objective optimal designs.

There are two ways of formulating the multiple-objective optimal design problems. One is based on a compound optimality criteria and the other is based on a constrained optimality criteria. The former one formulates a new concave real-valued function as the weighted sum of the multiple objectives. An attractive property of compound optimality is that the objective function maintains concavity property, which is critical for applying the celebrated equivalence theorem. For given weights, the corresponding optimal design can be derived through the new algorithm proposed by (Yang, Biedermann and Tang, 2013). A big drawback of this approach is the choice of weights. It does not in general have a meaningful interpretation.

The latter formulates the optimality problem as maximizing one objective function subject to all other objective functions satisfying certain efficiencies. The constrained optimization approach provides a clearer and more intuitive interpretation than the compound optimality approach. This has made it a popular method. However, the constrained optimization approach does not maintain the concave property. Consequently, there is no general approach of deriving constrained optimal design. Fortunately, there is a relationship between the two approaches. Based on the Lagrange multiplier theorem, (Clyde and Chaloner, 1996) generalized a result of (Cook and Wong, 1994) and showed the equivalence of the constrained optimization approach and the compound optimality approach. A numerical solution for the constrained design problem can be derived by using an appropriate compound optimality criteria. In fact, almost all numerical solutions for constrained design problems use this strategy. But the major challenge is how to find the corresponding weights for a given constrained optimality problem.

There are two approaches in the literature: the grid search approach and the sequential approach. For the grid search approach, the number of grid points increases exponentially with the number of objectives, and can be huge even for a moderate number of objectives.

For example, with four objectives and a grid size of 0.01 for each dimension of weights, the total number of grid points is well beyond 170000. Since the best design must be found for each of these, this becomes very quickly computationally infeasible. With three objectives, (Huang and Wong, 1998) proposed a sequential approach for finding the weights. The basic idea is to consider the objective functions in pairs and sequentially add more constraints. While this seems to have given reasonable answers in their examples, there lacks theoretical justification. Consequently this approach will generally not yield satisfied solution even for the three-objective optimal design problems.

The goal of this paper is to propose a novel algorithm for deriving optimal designs for multiple objective function. For a given constrained optimization problem, if the solution exists, we prove that new algorithm guarantees to find the desired weights. The new algorithm is also fast. For example, for a design with three constraints, the new algorithm can find a desired solution within 30 minutes with a laptop while the grid search approach will take more than 10 hours and the sequential approach fails to produce a desired solution.

This paper is organized as follows. In Section 2, we introduce the set up and necessary notation. The main results including characterization, convergence properties, implemention of the algorithm, as well as the computation cost are presented in Section 3. Applications to many different nonlinear models and different number of constrains, and comparisons with grid search and sequential approach are shown in Section 4. Section 5 provides a brief discussion, followed by an appendix containing the proofs.

2 Set up and Notation

We adapt the same notation as that of (Yang, Biedermann and Tang, 2013). Suppose we have a nonlinear regression model for which at each point $\mathbf{x}$ the experimenter observes a response Y. Here $\mathbf{x}$ could be a vector, and we assume that the responses are independent and follow some distribution from the exponential family with mean $\eta(\mathbf{x},\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the $(k \times 1)$ vector of unknown parameters. Typically, approximate designs are studied, i.e. designs of the form $\xi = \{(\mathbf{x}_i, \omega_i), i = 1, \ldots, m\}$ with support points $\mathbf{x}_i \in \mathcal{X}$ and weights $\omega_i > 0$, and $\sum_{i=1}^{m} \omega_i = 1$. Denote the original design space as $\mathcal{X}$. The set of all approximate designs on the design region $\mathcal{X}$ is denoted by Ξ.

Denote the information matrix of ξ as $\mathbf{I}_\xi$. Let $\Phi_0(\xi), \ldots, \Phi_n(\xi)$ be the values of $n+1$ smooth objective functions for design ξ. These objective functions are some real-valued functions of $\mathbf{I}_\xi$ which are formulated such that larger values are desirable. These objectives depend on the optimality criteria and the parameters of interest and different objective may have different parameters of interest. For example, $\Phi_0(\xi)$ can be the opposite number of the trace of inverse of the information matrix; $\Phi_1(\xi)$ can be the opposite number of the determinant of the inverse of the corresponding information matrix when the parameter of interest is restricted to the first two parameters (assuming there are more than two parameters).

Ideally, we hope we can find a ξ^* which can maximize $\Phi_0(\xi), \ldots, \Phi_n(\xi)$ simultaneously among all possible designs. Such ideal solution does not exist in general unless under some special situations. For example, only one parameter in the model. So it is infeasible to maximize all objectives simultaneously. One commonly used way of formulating the multiple objectives optimal design problems is constrained optimization approach, which specifies one objective as the primary criteria and maximizes this objective subject to the constraints on the remaining objectives (Cook and Wong, 1994; Clyde and Chaloner, 1996) . Formally, this approach can be written as

$$\underset{\xi \in \Xi}{\text{Maximize}}\ \Phi_0(\xi) \text{ subject to } \Phi_i(\xi) \geq c_i,\ i = 1, \ldots, n, \tag{1}$$

where $\mathbf{c} = (c_1, \ldots, c_n)$ are user-specified constants which reflect minimally desired levels of performance relative to optimal designs for these n objective functions. To make this problem meaningful, throughout this paper, we assume there is at least one design satisfying all the constraints, which means an optimal solution exists. The constrained optimization approach provides a clear and intuitive interpretation. It enables a user to specify the order of importance among the objective functions in a simple and meaningful way. This has made it a popular method.

Unfortunately, with the restricted optimality set up, there is no direct way of solving the constrained optimization problem. We have to solve (1) through the corresponding compound optimal design. Let

$$L(\xi, \mathbf{U}) = \Phi_0(\xi) + \sum_{i=1}^{n} u_i(\Phi_i(\xi) - c_i), \tag{2}$$

where $u_i \geq 0,\ i = 1, \ldots, n$. Let $\mathbf{U} = (u_1, \ldots, u_n)$. For a given $\mathbf{U}$, $L(\xi, \mathbf{U})$ maintains the concavity property without any restriction. This property is critically important for applying the celebrated equivalence theorem, which enables verification whether a given design is indeed optimal. More important, this property allows us to apply the newly developed algorithm OWEA. Thus deriving a design maximizing $L(\xi, \mathbf{U})$ is relatively easy once the $\mathbf{U}$ is given.

Notice that it is not recommended to use compound optimal design strategy directly for multiple objective optimal designs. The reason is that the choice of $\mathbf{U}$ is the main difficulty and it does not in general have a meaningful interpretation.

To establish the relationship between constrained optimal design and compound optimal design, we need the following assumptions, which are adapted from (Clyde and Chaloner, 1996) . Assume that

(A1) $\Phi_i(\xi)$, $i = 0, \ldots, n$ are concave on Ξ.

(A2) $\Phi_i(\xi)$, $i = 0, \ldots, n$ are differentiable and the directional derivatives are continuous on $\mathbf{x}$.

(A3) If ξ_n converges to ξ, then $\Phi_i(\xi_n)$ converges to $\Phi_i(\xi)$, $i = 0, \ldots, n$.

(A4) There is at least one design ξ in Ξ such that the constraints (1) are satisfied.

(Clyde and Chaloner, 1996) generalized a result of (Cook and Wong, 1994) and showed the equivalence of the constrained optimization approach and the compound optimality approach.

Theorem 1 *(Clyde and Chaloner, 1996). Under assumptions A1 to A4, ξ^* is optimal for constrained optimal design (1) if and only if there exists a non-negative vector $\mathbf{U}^* = (u_1^*, \cdots, u_n^*) \in \Re^n$, such that*

$$\xi^* = argmax_{\xi \in \Xi} L(\xi, \mathbf{U}^*), \Phi_i(\xi^*) \geq c_i \text{ for } i = 1, \cdots, n \text{ and } \sum_{i=1}^{n} u_i^*(\Phi_i(\xi^*) - c_i) = 0. \tag{3}$$

Theorem 1 provides necessary and sufficient condition for constrained optimal designs (1). It demonstrates that a numerical solution for the constrained design problem (1) can be derived by using an appropriate compound optimality criteria. In fact, almost all numerical solutions for constrained design problems use this strategy. The big challenge is how to find the desired $\mathbf{U}^*$ for a given constrained design problem (1). There are two approaches to handle this: the grid search approach and the sequential approach. Both approach consider the weighted optimal design, which is equivalent to compound optimal design. Let

$$\Phi_\lambda(\xi) = \sum_{i=0}^{n} \lambda_i \Phi_i(\xi), \tag{4}$$

where $\lambda = (\lambda_0, \ldots, \lambda_n)$, $\lambda_0 > 0$, $0 \leq \lambda_i < 1$, $i = 1, \ldots, n$ with $\sum_{i=0}^{n} \lambda_i = 1$. Clearly $\Phi_\lambda(\xi)$ is just a normalized form of $L(\xi, \mathbf{U})$. For given λ, $\Phi_\lambda(\xi)$ also enjoys the concave property as $L(\xi, \mathbf{U})$ does. So deriving a weighted optimal design can be based on the some standard algorithm or the newly developed algorithm OWEA.

As we discuss in the introduction section, both grid search and the sequential approach (we shall give detailed description later) have their own problems. Consequently they cannot serve as a general solution for the constrained optimal design problem (1). How can we develop a general and efficient algorithm for the important but largely unsolved problem? The first step is to characterize $\mathbf{U}^*$ in Theorem 1.

3 Characterization

For deriving theoretical results purpose, we need to have two assumptions. The first one is

$$\Phi_0 \text{ is a strong concave function on information matrices.} \tag{5}$$

The strong concave property means the optimal design is unique in term of information matrix, i.e., if ξ_1^* and ξ_2^* both are optimal designs for $L(\xi, \mathbf{U}^0)$(2) with a fixed Lagrange multiplier $\mathbf{U}^0$, then the two information matrices of ξ_1^* and ξ_2^* are identical. Assumption (5) is not restrict. Many optimality objective functions satisfy this assumption. For example, D-, A-, E-, and general ϕ_p-optimality criteria for full parameters satisfy this assumption. Let ξ^* be the optimal design for a constrained optimal design problem (1). By Theorem 1, ξ^* is also an optimality solution of a compound optimal design problem (2). Let $\mathbf{U}^* = (u_1^*, \ldots, u_n^*)$ be the Lagrange multiplier of the compound optimal design problem.

In a compound optimal design problem (2), each $u_i > 0$ without upper bound. For an algorithm searching for $\mathbf{U}^*$, establish the convergence property of the algorithm is challenging when the search space is infinite. Thus our second assumption is

$$u_i^* \in [0, N_i) \text{ where } N_i \text{ is pre-specified, } i = 1, \ldots, n. \tag{6}$$

This assumption is equivalent to the grid size in a weighted optimal design problem (4). Both grid search approach and sequential approach need to choose a grid size. Let the grid size be ϵ, then it means $0 \leq u_i \leq \frac{1-\epsilon}{\epsilon} < \frac{1}{\epsilon}$ for the equivalent compound optimal design (2). We can always choose some reasonable large numbers N_i's such that Assumption (6) is satisfied.

A constraint Φ_i is called active if $u_i^* > 0$; otherwise the constraint will be regarded as inactive. For easy presentation, we denote $\xi_{\mathbf{U}}$ as a design which maximizes the Lagrange function $L(\xi, \mathbf{U})$ for a given weight vector $\mathbf{U} = (u_1, \cdots, u_n)$ and $\hat{\Phi}_i(\xi)$ as $\Phi_i(\xi) - c_i$, $i = 1, \ldots, n$. Before we characterize $\mathbf{U}$ in Theorem 1, we first give an overview of the new algorithm. The detailed description is given in Section 4. The overview can help us to characterizer some desired properties of compound optimal designs of (2).

3.1 Overview of the new algorithm

The new algorithm is designed to search for a satisfied $\mathbf{U}^*$ from the most easiest case to the most complex case:

$$\mathbf{U}^* \text{ is a zero vector} \longrightarrow \text{elements in } U^* \text{ are all nonzero.}$$

In other words, the algorithm will go through all the possible cases:

$$\text{all constraints are inactive} \longrightarrow \text{all constraints are active}$$

if needed.

Now consider that the constrained optimal design problem have a active constraints. Without losing generality, suppose these active constrains are $\Phi_1, \cdots, \Phi_a$. In other words, our efforts now are on finding a weight vector $\mathbf{U} = (u_1, \cdots, u_a, u_{a+1}, \cdots, u_n)$ where $u_1, \cdots, u_a$ are positive and $u_{a+1}, \cdots, u_n$ are zero and hopefully $\xi_{\mathbf{U}}$ will satisfy the sufficient condition.

To search for satisfied values for $u_1, \cdots, u_a$, the algorithm will use bisection process for all elements $u_1, \cdots, u_a$ through a recurrent process. The rest element $u_{a+1}, \cdots, u_n$ in weight vector $\mathbf{U}$ will be fixed at 0 during the bisection process. Denote bisection result of $\mathbf{U}$ by $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$. Then for any $i \in \{1, \cdots, a\}$, u_i^* will satisfy the following property:

$$\begin{aligned}
&\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) > 0, \text{ then } u_i^* = 0; \\
&\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) < 0, \text{ then } u_i^* = N_i; \\
&\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0, \text{ then } u_i^* \in [0, N_i].
\end{aligned} \tag{7}$$

This property will be quoted frequently in the later theorems.

For example, take $a = 2$, which means only u_1 and u_2 are supposed to be nonzero. In this case, the algorithm first fixes u_2 as $u_2^0 = \frac{0+N_2}{2}$. Then the value for u_1 will be updated to

u_1^0 using bisection and u_1^0 will satisfy Property (7) with $\mathbf{U}^0 = (u_1^0, u_2^0, 0, \cdots, 0)$. Now check $\hat{\Phi}_2(\xi_{U^0})$. If $\hat{\Phi}_2(\xi_{\mathbf{U}^0}) \neq 0$, adjust the value for u_2 through one time bisection to get u_2^1 such that $\hat{\Phi}_2(\xi_{\mathbf{U}^1})$ is closer to 0. For the new fixed $u_2 = u_2^1$, again update u_1 to u_1^1 using bisection to make u_1^1 satisfy Property (7) with $U^1 = (u_1^1, u_2^1, 0, \cdots, 0)$. Check $\hat{\Phi}_2(\xi_{U^1})$ and update u_2 to u_2^2 if $\hat{\Phi}_2(\xi_{U^1}) \neq 0$. $\cdots$. Continue this process until a satisfied $\mathbf{U}^* = (u_1^*, u_2^*, 0, \cdots, 0)$ is found which guarantees that u_1^* and u_2^* both satisfy Property (7).

For a general a active constraints case, similar to $a = 2$ case, we first fix u_a as $u_a^0 = \frac{0+N_a}{2}$. Similar to the recurrent process mentioned for 2 active constraints case, derive the corresponding values $u_1^0, \cdots, u_{a-1}^0$ for the element u_1 to u_{a-1} using bisections approach such that they satisfy Property (7) with $\mathbf{U}^0 = \{u_1^0, \cdots, u_a^0, 0, \cdots, 0\}$. Check whether $\hat{\Phi}_a(\xi_{\mathbf{U}^0}) = 0$ and update u_a to u_a^1. $\cdots$. Continue this process until a desired $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$ is found with all $u_1^*, \cdots, u_a^*$ satisfied Property (7).

To guarantee the bisection technique is valid and the desired Property (7) can be achieved for $u_1, \cdots, u_a$ through the bisection process, we need to characterize the property of the multiplier $\mathbf{U}$. The characterizations in this section allow us to propose a new algorithm which guarantees the convergence and speed.

3.2 Properties

Theorem 2 *For any $a \in \{1, \cdots, n\}$, $S \subsetneq \{1, \cdots, n\} \backslash \{a\}$ and $S' = \{1, \cdots, n\} \backslash (S \bigcup \{a\})$, define $\mathbf{U}_S = \{u_i | i \in S\}$ and $\mathbf{U}_{S'} = \{u_i | i \in S'\}$. Then $\hat{\Phi}_a(\xi_{\mathbf{U}})$ is a non-decreasing function of u_a if $\mathbf{U}_{S'}$ is pre-fixed and $\mathbf{U}_S$ satisfies one of the following two conditions:*

$$\begin{aligned} &\hat{\Phi}_i(\xi_{\mathbf{U}}) \geq 0 \text{ and } u_i \hat{\Phi}_i(\xi_{\mathbf{U}}) = 0 \text{ for } i \in S_1, \text{ or} \\ &u_i = N_i \text{ and } \Phi_i(\xi_{\mathbf{U}}) < 0 \text{ for } i \in S_2, \end{aligned} \tag{8}$$

where $S_1 \cup S_2 = S$ and $S_1 \cap S_2 = \emptyset$ and $\mathbf{U}$ is the combination of $\mathbf{U}_S$, u_a, and $\mathbf{U}_{S'}$ by their corresponding indexes.

Condition (8) implies that $u_i, i \in S$ satisfy Property (7). Suppose there are a active constraints and they are $\Phi_1, \cdots, \Phi_a$. When we search for the proper value of u_i $(i \leq a-1)$, $u_{i+1}, \cdots, u_a$ and the zero-element $u_{a+1}, \cdots, u_n$ can be regarded as fixed, which correspond to $\mathbf{U}_{S'}$ in theorem. And since it is a recurrent process, for $u_1, \cdots, u_{i-1}$, the value will be updated first according to the value assigned to u_i each time and fixed $u_{i+1}, \cdots, u_n$. Thus $(u_1, \cdots, u_{i-1})$ is $\mathbf{U}_S$ in this case. After $u_1, \cdots, u_{i-1}$ being updated for the given u_i, $\hat{\Phi}_i(\xi_{\mathbf{U}})$ should be a monotone increasing function of u_i by Theorem 2. Due to the monotone property, three cases may occur when we search for u_i:

Case 1 $\hat{\Phi}_i(\xi_{\mathbf{U}}) = 0$ and $u_i \in [0, N_i]$;

Case 2 $\hat{\Phi}_i(\xi_{\mathbf{U}}) < 0$ and $u_i = N_i$;

Case 3 $\hat{\Phi}_i(\xi_{\mathbf{U}}) > 0$ and $u_i = 0$.

The three possible cases are equivalent to Property 7. Under all these possible cases that may occur when the bisection technique is applied to the former elements, Theorem 2 makes it clear that the monotone increasing property holds for the next element to which the bisection technique is applied.

Now suppose the active constraints are Φ_i with $i \in S \subseteq \{1, \cdots, n\}$. A weight vector $\mathbf{U}_S^*$ for active constraints can be found through the bisection technique. One can always construct a complete weight vector $\mathbf{U}^* = (u_1^*, \cdots, u_n^*)$ as follows:

For any $i \in \{1, \cdots, n\}$

- If $i \in S$, take u_i^* as the corresponding value in $\mathbf{U}_S^*$;
- If $i \notin S$, $u_i^* = 0$.

For simplicity, we denote such constructed full weight vector $\mathbf{U}$ as $\{\mathbf{U}_S, 0\}$.

Theorem 3 *Define $S \subsetneq \{1, \cdots, n\}$ as the active constraints indexes set. For two non-zero value sets $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$ of the corresponding weight vector $\mathbf{U}_S$, let $\mathbf{U}^0 = \{\mathbf{U}_S^0, 0\}$ and $\mathbf{U}^1 = \{\mathbf{U}_S^1, 0\}$, then $\xi_{\mathbf{U}^0}$ will be equivalent to $\xi_{\mathbf{U}^1}$, i.e., they have the same information matrix, if the two designs both satisfy*

$$\hat{\Phi}_i(\xi) = 0, i \in S. \tag{9}$$

Suppose $\Phi_1, \cdots, \Phi_a$ are active constraints. Theorem 3 shows all the possible weight vectors, that satisfy $\hat{\Phi}_i(\xi_{\mathbf{U}}) = 0$, $i \in \{1, \cdots, a\}$, are equivalent. Thus if we find $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$ with $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for $i \in \{1, \cdots, a\}$, $\mathbf{U}^*$ can represent all the possible satisfied weight vectors since they are all equivalent. For such $\mathbf{U}^*$, if $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \geq 0$ for $i = 1, \cdots, n$; then assumption is proper and $\mathbf{U}^*$ will be the desired weight vector; otherwise the assumption is not valid and two cases need to be considered:

Case 1 There are still a active constraints but we need to pick another combination of constraints of size a and re-do the searching process.

Case 2 If all combinations of sized a constraints have been tested and a desired $\mathbf{U}^*$ cannot be found, then it implies the constrained optimal design problem has more than a active constraints and $a + 1$ active constraints cases should be considered.

However, the bisection technique may return a weight vector with some elements, say i-th element, taking value at lower bound 0 or upper bound N_i, while the corresponding $\hat{\Phi}_i \neq 0$. In this situation, the following theorem guarantees that the assumed active constrains set is not valid and we can move to a new active constraints set according to the two cases mentioned above.

Theorem 4 *For any $S \subset \{1, \cdots, n\}$, suppose that $\mathbf{U}^0 = \{\mathbf{U}_S^0, 0\}$ satisfies the following two conditions*

$$\begin{aligned} &(i)\ \hat{\Phi}_i(\xi_{\mathbf{U}^0}) \geq 0 \text{ for } i \in S_1 \text{ and } \sum_{i \in S_1} u_i \hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0. \\ &(ii)\ \hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0 \text{ and } u_i = N_i \text{ for } i \in S_2. \end{aligned} \tag{10}$$

where $S_1 \cup S_2 = S$ and $S_1 \cap S_2 = \emptyset$. If there exists at least one element in S, say i, such that $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) \neq 0$, then there does not exist a positive value set $\mathbf{U}_S^+ = \{u_i \in (0, N_i) | i \in S\}$, such that $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for $i \in S$, where $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$.

Now we are ready to present the new algorithm.

4 Algorithm

For a given constrained optimal design problem (1), the new algorithm is to find the desired $\mathbf{U}^*$. In each step, we need to derive an optimal design for a compound optimal design problem (2) with $\mathbf{U}$ being given. We first introduce such algorithm.

4.1 Deriving Compound Optimal Design with given U

(Yang, Biedermann and Tang, 2013) proposed the optimal weight exchange algorithm (OWEA), a general and efficient algorithm for deriving optimal designs. OWEA can be applied to commonly used optimality criteria regardless of the parameters of interest and also enjoys high speed. This algorithm was originally designed for one objective optimal design problems. Fortunately, OWEA can be extended for deriving $\xi_{\mathbf{U}} = argmax_{\xi} L(\xi, \mathbf{U})$ where $\mathbf{U}$ is given.

Since all elements in $\mathbf{U}$ are nonnegative, $L(\xi, \mathbf{U}) = \Phi_0(\xi) + \sum_{i=1}^{n} u_i(\Phi_i(\xi) - c_i)$ can be regarded as a new optimal criteria. For a design $\xi = \{(\mathbf{x}_1, w_1), \cdots, (\mathbf{x}_{m-1}, w_{m-1}), (\mathbf{x}_m, w_m)\}$, let $X = (\mathbf{x}_1, \cdots, \mathbf{x}_m)^T$ and $W = (w_1, \cdots, w_{m-1})^T$. The following algorithm follows the similar procedure as that of OWEA in (Yang, Biedermann and Tang, 2013).

Step 1 Set $t = 0$, let the initial design set X^0 take $2k$ design points uniformly from the design space and the corresponding weight to be $1/2k$ for each point.

Step 2 Derive the optimal weight vector W^t for a fixed sample points set X^t.

Step 3 For $\xi^t = (X^t, W^t)$, denote directional derivative of $L(\xi, \mathbf{U})$ at $\mathbf{x}$ as $d_{\mathbf{U}}(\mathbf{x}, \xi^t)$, where $\mathbf{x}$ is any design point from the design space $\mathcal{X}$. The explicit expression can be found in (Yang, Biedermann and Tang, 2013).

Step 4 For a small prefixed value $\Delta > 0$, if $\max_{\mathbf{x}\in\mathcal{X}} d_{\mathbf{U}}(\mathbf{x}, \xi^t) \leq \Delta$, ξ^t can be regarded as the optimal design. If $d_{\mathbf{U}}(\mathbf{x}, \xi^t) > \Delta$ for some design point $\mathbf{x}$, let $X^{t+1} = X^t \bigcup \hat{\mathbf{x}}_t$ where $\hat{\mathbf{x}}_t = \underset{\mathbf{x}\in\mathcal{X}}{argmax}\ d_{\mathbf{U}}(\mathbf{x}, \xi^t)$. Go through Step 2 to Step 4 again with new X^{t+1}.

In Step 2, the optimal weight vector $\hat{W}$ can be found by Newton's method based on the first derivative and second derivative of $L(\xi, \mathbf{U})$ respect to the weight vector W. These derivatives can be derived using (11) and the formula in the Appendix of (Yang, Biedermann and Tang, 2013).

$$\begin{aligned} \frac{\partial \Phi_\lambda(\xi)}{\partial W} &= \frac{\partial \Phi_0(\xi)}{\partial W} + \sum_{i=1}^{n} u_i \frac{\partial \Phi_i(\xi)}{\partial W}; \\ \frac{\partial^2 \Phi_\lambda(\xi)}{\partial W W^T} &= \frac{\partial^2 \Phi_0(\xi)}{\partial W W^T} + \sum_{i=1}^{n} u_i \frac{\partial^2 \Phi_i(\xi)}{\partial W W^T}. \end{aligned} \tag{11}$$

Based on the exact same argument as Yang, Biedermann and Tang (2013), this algorithm converges to an optimal design maximizing $L(\xi, \mathbf{U})$. We use the extended OWEA to derive $\xi_{\mathbf{U}}$. Now we are ready to present the main algorithm which is to search the satisfied $\mathbf{U}^*$.

4.2 The main algorithm

The strategy of the algorithm is to search from the simplest case (no constraint is active) to the most complicated case (all constraints are active). The algorithm can be described as following:

Step 1 Set $a = 0$, derive $\xi^* = \underset{\xi}{argmax}\ \Phi_0(\xi)$ and check whether $\Phi_i(\xi^*) \geq c_i$ for $i = 1, \cdots, n$. If all constrains are satisfied, stop and ξ^* is the desired design. Otherwise set $a = 1$ and go to Step 2.

Step 2 Set $i = 1$, consider $\xi^* = \underset{\xi}{argmax}\ \Phi_0(\xi) + u_i\Phi_i(\xi)$. Adjust the value of u_i using the bisection technique on $[0, N_i]$ to obtain u_i^* such that $\hat{\Phi}_i(\xi^*) = 0$. During the bisection process, the upper bound, instead of the median, of the final bisection interval will be picked as the right value for u_i^*. If $\hat{\Phi}_i(\xi^*) > 0$ when $u_i = 0$, set $u_i^* = 0$. If $\hat{\Phi}_i(\xi^*) < 0$ when $u_i = N_i$, set $u_i^* = N_i$. For $\xi^* = \underset{\xi}{argmax}\ \Phi_0(\xi) + u_i^*\Phi_i(\xi)$, check whether $\hat{\Phi}_j(\xi^*) \geq 0$ for $j = 1, \cdots, n$. If all constraints are satisfied, stop and ξ^* is the desired design; otherwise change i to $i + 1$ and repeat this process. After $i = n$ is tested and no desired ξ^* is found, then set $a = 2$ and proceed to Step 3.

Step 3 Find all subsets of $\{1, \cdots, n\}$ of size a, choose one out of these subsets. Denote it as S.

Step 4 Let $(s_1, \ldots, s_a)$ be the indexes of the elements in $\mathbf{U}_S$. To find the right value $\mathbf{U}_S^*$ for $\mathbf{U}_S$, we follow a recurrent process. For each time a given value of u_{s_a}, first use bisection technique to find the corresponding $u_{s_1}, \cdots, u_{s_{a-1}}$. The full weight vector $\mathbf{U}$ can be constructed with $u_{s_1}, \cdots, u_{s_a}$ by setting all the other weight elements in $\mathbf{U}$ as 0's, which we later denote by $\mathbf{U} = \{\mathbf{U}_S, 0\}$. Then adapt the value of u_{s_a} as follows :

- If $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) > 0$ when u_{s_a} is assigned as 0, set $u_{s_a}^* = 0$.
- If $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) < 0$ when u_{s_k} is assigned as N_a, set $u_{s_a}^* = N_a$.
- Otherwise use the bisection technique to find $u_{s_a}^*$ such that $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) = 0$.

Record $u_{s_a}^*$ and the corresponding values for $\{u_{s_1}^*, \cdots, u_{s_{a-1}}^*\}$ as U_S^*. For the bisection process in each dimension, the upper bound of the final bisection interval will be picked as the right value for the corresponding element in weight vector $\mathbf{U}_S^*$. Then the full weight vector $\mathbf{U}^*$ can be constructed using $\mathbf{U}^* = \{\mathbf{U}_S^*, 0\}$.

Step 5 For the U_S^* and $\xi_{\mathbf{U}^*}$ derived in Step 4, check $\hat{\Phi}_i(\xi_{\mathbf{U}^*})$, $i = 1, \ldots, n$. If all constraints are satisfied, stop and $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise, pick another a-element subset in Step 3, and go through Step 4 to Step 5 again. If all a-element subsets are tested, go to Step 6.

Step 6 Change a to $a + 1$, go through Step 3 to Step 5, until $a = n$. If no suitable design $\xi_{\mathbf{U}^*}$ is found, the implication is that there is no solution for the constrained optimal design (1).

We demonstrate this algorithm through an optimal design problem with two constraints. Denote the target objective function by Φ_0 and two constrained objective functions by Φ_1 and Φ_2. The algorithm will search for a desired weight vector $\mathbf{U}^* = (u_1^*, u_2^*)$ and desired design $\xi_{\mathbf{U}^*}$ according to the following process:

Step 1 Suppose there is no active constraint, then $\mathbf{U}^*$ in this case will be $(0,0)$ and $\xi_{\mathbf{U}^*}$ is also an optimal design for Φ_0. If $\xi_{\mathbf{U}^*}$ satisfies all the constraints, then $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise go to Step 2.

Step 2 Suppose there is one active constraint. First suppose Φ_1 is active. Derive u_1^* through bisection technique such that $\hat{\Phi}_1(\xi_{\mathbf{U}^*}) = 0$, where $\mathbf{U}^* = (u_1^*, 0)$. If $\xi_{\mathbf{U}^*}$ satisfies all the constraints, $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise suppose Φ_2 is active and repeat this process. If both fail to find the desired $\xi_{\mathbf{U}^*}$, that means there are more than one active constraint. Go to Step 3.

Step 3 Now suppose all constraints are active. Derive $\mathbf{U}^* = (u_1^*, u_2^*)$ through bisection technique such that $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for $i = 1, 2$. If such $\mathbf{U}^*$ can be derived, then $\xi_{\mathbf{U}^*}$ is the desired design. If it fails to produce a satisfied $\mathbf{U}^*$, there are two possible reasons:

Case 1 The predefined upper bound vector N_1 and N_2 are not proper. The true u_i^* fall out of the interval $[0, N_i)$ for at least one of i's, $i = 1, 2$,

Case 2 There is no solution for the constrained optimal design problem.

4.3 Convergence and Computational Cost

Whether an algorithm is successful mainly depends on two properties: convergence and computational cost. We first establish the convergence of the proposed algorithm.

Theorem 5 *For the constrained optimal design problem (1), under Assumptions (5) and (6), the proposed algorithm converges to* ξ^*.

Proof. Since there exists an optimal solution for the constrained optimal design problem (1), there exists an active constraints set (it could be empty set, which means no active constraints). The new algorithm will search for this active constraints set and identify the Lagrange multiplier of the corresponding compound optimal design problem. The new algorithm starts from the simplest case, i.e., there is no active constraints, to most complex case, i.e., all constraints are active.

For each assumed active constraints set S, by Theorem 2, the algorithm procedure utilizing the bisection technique to guarantee that the derived vector $\mathbf{U}^* = \{\mathbf{U}_S^*, 0\}$ satisfies the two conditions in (10). If $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \neq 0$ for some $i \in S$, Theorem 4 guarantees that there is no positive value set $\mathbf{U}_S^+$ within the given intervals such that $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for all $i \in S$ where $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$. This means S cannot be the true active constraints set. Otherwise it contradicts to Assumption (6). On the other hand, if $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for all $i \in S$ but $\hat{\Phi}_{i'}(\xi_{\mathbf{U}^*}) < 0$ for some $i' \in \{1, \cdots, n\} \backslash S$, Theorem 3 guarantees that $\hat{\Phi}_i'(\xi_{\mathbf{U}'}) < 0$ for any vector $\mathbf{U}' = \{\mathbf{U}_S', 0\}$ satisfying $\hat{\Phi}_i(\xi_{\mathbf{U}'}) = 0$ for all $i \in S$. This also means S cannot be the true active constraints set.

Table 1: Comparison of Caculational Cost

	Three Objectives		Four Objectives	
Mesh Grid Size	0.01	0.001	0.01	0.001
Grid Search	5050	500500	171700	167167000
New Algorithm	265	525	4320	12058

Since the new algorithm goes through all possible active constraints combinations, a desireded $\mathbf{U}^*$, i.e., $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for all $i \in S$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \geq 0$ for all $i \in \{1, \cdots, n\}\backslash S$, must be found. Otherwise, it means none of the constraints combinations is active. This contradicts to the fact that there is an active constraints set.

For the desired $\mathbf{U}^*$, let $\xi^* = argmax_\xi L(\xi, \mathbf{U}^*)$. By Theorem 1, ξ^* is the optimal design of the constrained optimal design problem (1). ■

Next we shall compare the computational cost of the new algorithm with those of the grid search and the sequential approach. Both the grid search and the sequential approach are based on weighted optimal design problem (4), which is equivalent to a compound optimal design problem with $u_i = \frac{\lambda_i}{\lambda_0}$, $i = 1, \ldots, n$. All three approaches are based on identifying a satisfied multiplier of a compounded optimal design problem and the computational cost of each approach is proportional to the number of multiplier the approach tests.

The grid search approach considers all possible combinations of $\lambda_1, \cdots, \lambda_n$ on $[0, 1]^n$ with given mesh grid size. The combination must satisfy that $\sum_{i=1}^n \lambda_i < 1$ and λ_0 is set as $1 - \sum_{i=1}^n \lambda_i$. Suppose the grid size is ϵ in a grid search. Let T_G be the number of all possible combinations. Direct computation shows that

$$T_G = \sum_{k=0}^{n} \binom{n}{k} \binom{\lfloor \frac{1}{\epsilon} \rfloor - 1}{k} = \binom{n + \lfloor \frac{1}{\epsilon} \rfloor - 1}{n}, \tag{12}$$

where $\lfloor . \rfloor$ refers to floor function.

For the new algorithm, since $u_i = \frac{\lambda_i}{\lambda_0}$, the upper bound of the corresponding u_i is $1/\epsilon$. To guarantee the new algorithm has at least the same accuracy (ϵ) on interval $[0, 1/\epsilon]$ as that of grid search, one needs $\lceil -2log_2\epsilon + 2 \rceil$ times bisection technique. Here $\lceil . \rceil$ refers the ceiling function. Let T_L be the number of times compound optimal designs calculated during the searching process, then

$$T_L = \sum_{k=0}^{n} \binom{n}{k} \lceil -2log_2\epsilon + 2 \rceil^k = \lceil -2log_2\epsilon + 3 \rceil^n. \tag{13}$$

As for the sequential approach, the computational cost is significantly less than those of the grid search and the new algorithm. However, as we will demonstrate in the next section, the sequential approach in general cannot find a desired solution.

Tables 1 and 2 shows the comparison of computational cost between new algorithm and grid search under different grid sizes and different numbers of constraints.

5 Numerical Examples

In this section, we will compare the performance (accuracy and the computing time) of the new algorithm, the grid search and the sequential approach. The sequential approach was introduced in (Huang and Wong, 1998). This approach first reorder $\Phi_0, \cdots, \Phi_n$ as $\Phi_{s_1}, \cdots, \Phi_{s_{n+1}}$ according to a robustness technique. In this paper, we test all possible orders and pick up the best design. Certainly it includes the special pick in (Huang and Wong, 1998). The constraint for target optimality Φ_0 can be regarded as $c_0 = 0$ and then combine c_0 with the original constraints vector $c = (c_1, \cdots, c_n)$. For newly constructed $c^* = (c_0, \ldots, c_n)$, reorder it as $(c_{s_1}, \cdots, c_{s_{n+1}})$. Then the sequential procedure for finding the corresponding compound optimal design with the specified order $\{s_1, \cdots, s_{n+1}\}$ can be described as follows:

Step 1 If $\Phi_0 \in \{\Phi_{s_1}, \Phi_{s_2}\}$, say $\Phi_0 = \Phi_{s_1}$. Consider constrained optimal design problem

$$\text{Maximize } \Phi_0 \text{ while } \Phi_{s_2} \geq c_{s_2}.$$

If not, consider constrained optimal design problem

$$\text{Maximize } \Phi_{s_2} \text{ while } \Phi_{s_1} \geq c_{s_1}.$$

Finding the weight vector in the weighted optimal design problem corresponding to the specified constrained optimal design problem using the grid search with a pre-fixed grid size. Denote the weight vector by $(1-\beta_2, \beta_2)$. Construct a new objective function $\Phi_{\{s_1,s_2\}}(\xi) = \frac{(1-\beta_2)\Phi_{s_1}(\xi)+\beta_2\Phi_{s_2}(\xi)}{(1-\beta_2)\Phi_{s_1}(\xi_{s_1,s_2})+\beta_2\Phi_{s_2}(\xi_{s_1,s_2})}$, where ξ_{s_1,s_2} is optimal design for $(1-\beta_2)\Phi_{s_1}(\xi) + \beta_2\Phi_{s_2}(\xi)$. If $n \geq 2$, set $k = 3$.

Step 2 For the newly constructed objective function, consider weighted design problem $(1 - x)\Phi_{\{s_1,\cdots,s_{k-1}\}} + x\Phi_{s_k}$. Change the value of x by grid search on $[0, 1]$ with given grid size. If $\Phi_0 \in \{\Phi_{s_1}, \cdots, \Phi_{s_k}\}$, choose a proper value x such that the corresponding weight design maximizes Φ_0 while guarantees $\Phi_{s_i} \geq c_i$ for $i = 1, \cdots, k$. If not, choose a proper value x such that the corresponding weighted optimal design maximizes Φ_{s_k} while guarantees $\Phi_{s_i} \geq c_i$ for $i = 1, \cdots, k-1$. Denote this value as β_k. If all the possible value for x fails to satisfy the constraints for $\Phi_{s_1}, \cdots, \Phi_{s_k}$, that indicates the sequential approach fails with the specified order. Then quit the algorithm.

Construct new objective function

$$\Phi_{\{s_1,\cdots,s_k\}}(\xi) = \frac{(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi) + \beta_k\Phi_{s_k}(\xi)}{(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi_{s_1,\cdots,s_k}) + \beta_k\Phi_{s_k}(\xi_{s_1,\cdots,s_k})},$$

where $\xi_{s_1,\cdots,s_k}$ is optimal design for $(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi) + \beta_k\Phi_{s_k}(\xi)$. Set $k = k+1$ and repeat Step 2, until $k = n+1$.

Step 3 Transfer $\Phi_{\{s_1,\cdots,s_{n+1}\}}(\xi)$ back to $\sum_{i=0}^{n} \lambda_i\Phi_i(\xi)$ using scalar change. Then $\sum_{i=0}^{n} \lambda_i\Phi_i(\xi)$ will be the weighted optimal design problem found for constrained design problem with the sequential approach based on the specified order.

For the grid search, weighted optimal design $\xi_\Lambda = \underset{\xi}{\text{argmax}} \sum_{i=0}^{n} \lambda_i \Phi_i(\xi)$ will be considered. All combinations of $\Lambda = (\lambda_0, \cdots, \lambda_n)^T$ will be checked using multi-dimensional grid search on $[0, 1]$ with constraint $\sum_{i=0}^{n} \lambda_i = 1$. Among all weighted optimal designs ξ_Λ, ξ^*, which maximizes Φ_0 while guarantee that $\Phi_i \geq c_i$ for $i = 1, \cdots, n$, is selected. Then ξ^* is regarded as an optimal design for the multiple-objective optimal design problem.

All three approaches utilize the OWEA algorithm to derive optimal designs for given weighted optimal design problems. For all examples, the design space has been discretized uniformly into 1000 design points. The cut-off value for checking optimality in $L(\xi, \mathbf{U})$ for given $\mathbf{U}$ was chosen to be $\Delta = 10^{-6}$. All other set ups of OWEA are the same as those of (Yang, Biedermann and Tang, 2013). For new algorithm and grid search, we require the algorithms to produce the best possible design while guarantee that the constraints are exactly satisfied. For sequential approach, since it doesn't guarantee to produce a proper design and may fail during the searching process, a tolerance value $\epsilon = 0.01$ is set up. That means during the sequential approach process, if a design ξ_0 have $\Phi_i(\xi_0) \geq c_i - \epsilon$ for some i, the design ξ_0 will still be regarded as a proper design which satisfies the constraint for objective function Φ_i. The grid size is 0.01 for all the examples in this section. The pre-specified upperbound N in the new algorithm is 100. All the algorithms are implemented in SAS software on a Lenovo laptop with Intel Core 2 duo CPU 2.27 HZ.

5.1 Three-objective Optimal Designs

In this subsection, we shall compare the performance of the grid search, the sequential approach, and the new algorithm in term of deriving optimal designs with three objectives.

Example I Consider the nonlinear model given by

$$y = \beta_1 e^{-\theta_1 x} + \beta_2 e^{-\theta_2 x} + \epsilon. \tag{14}$$

This model is commonly used to compare the progression of a drug between different compartments. Here y denotes the concentration level of the drug in compartments, x denotes the sampling time, and ϵ is assumed to follow normal distribution with mean zero and variance σ^2. In a PK/PD study, (Notari, 1980) used Model (14) to model the concentration of a drug taken at different time. The estimates of the parameters are $\theta_0 = (\theta_1, \theta_2, \beta_1, \beta_2) = (1.34, 0.13, 5.25, 1.75)$. Under these parameter estimations, (Huang and Wong, 1998) studied three-objective optimal design with design space $x \in [0, 15]$.

Let $B = \text{diag}\{\frac{1}{\theta_1^2}, \frac{1}{\theta_2^2}, \frac{1}{\beta_1^2}, \frac{1}{\beta_2^2}\}$; $W = \int_2^{10} f(x) f^t(x) v(dx)$, where $f(x)$ is the linearized function of the model function using Taylor expansion at θ_0^T; $\xi_0^* = argmin_\xi tr(I^{-1}(\xi)B)$; $\xi_1^* = argmin_\xi |I^{-1}(\xi)|$; and $\xi_2^* = argmin_\xi tr(I^{-1}(\xi)W)$. The three objective functions can be written as follow:

$$\Phi_0(I(\xi)) = -\frac{tr(I^{-1}(\xi)B)}{tr(I^{-1}(\xi_0^*)B)},$$
$$\Phi_1(I(\xi)) = -(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_1^*)|})^{\frac{1}{4}}, \text{ and}$$
$$\Phi_2(I(\xi)) = -\frac{tr(I^{-1}(\xi)W)}{tr(I^{-1}(\xi_2^*)W)}.$$

Define $\text{Effi}_{\Phi(\xi)} = -\frac{1}{\Phi(I(\xi))}$. Clearly $\text{Effi}_{\Phi_i(\xi)}$, $i = 0, 1, 2$ are consistent with the definitions of efficiency of design ξ under the corresponding optimality criteria. For example, $\text{Effi}_{\Phi_1(\xi)}$ refers the D-efficiency. Such definition will be used in the subsequent examples.

The three-objective optimal design problem considered in (Huang and Wong, 1998) is

$$\begin{aligned} &\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)} \\ &\text{subject to} \quad \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.9, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.8. \end{cases} \end{aligned}$$

Notice that the constraints $\text{Effi}_{\Phi_1(\xi)} \geq 0.9$ and $\text{Effi}_{\Phi_2(\xi)} \geq 0.8$ are obviously equivalent to $\Phi_1(I(\xi)) \geq -10/9$ and $\Phi_2(I(\xi)) \geq -5/4$, respectively. In the subsequent examples, we will use the similar efficiency setup without specifying their equivalence to the corresponding objective functions.

The efficiencies of ξ_1^*, ξ_2^*, and ξ_3^* under each of the three objective functions are shown in Table 2. Clearly the optimal design based on one single optimal criteria has bad performance under other optimal criteria. These efficiencies are consistent with the corresponding efficiencies provided in Table 4 of (Huang and Wong, 1998). The new algorithm is applied to the three-objective optimal design problem. With the new algorithm, the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 4.2053\Phi_1 + 2.5085\Phi_2.$$

The efficiencies of the derived constrained optimal design ξ^* are also shown in Table 2. It shows that ξ^* has high efficiency on Φ_0 while guarantees the other two efficiencies are above the acceptable level.

The grid search and the sequential approach are also applied to this optimal design problem. The sequential result is also consistent with that of (Huang and Wong, 1998). Table 3 shows the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm.

It shows that the three approaches are essentially equivalent. The sequential approach gains highest efficiency on Φ_0 by sacrificing a little bit on constrained efficiencies. New algorithm and grid search have slightly drop on target efficiency to guarantee that the two constraints are exactly satisfied. The sequential approach is faster. However, the computational time in the table for sequential approach is just for one possible order. In many cases, one may need to check many possible orders to produce a satisfied solution. Thus the

Table 2: Example I: the relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1	0.7315	0.7739
ξ_1^*	0.6677	1	0.5576
ξ_2^*	0.6959	0.4166	1
ξ^*	0.8692	0.9000	0.8001

Table 3: Example I: relative efficiencies of constrained optimal designs based on different techniques

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Time Cost (Seconds)
Grid Search	0.8658	0.9009	0.8000	1834
Sequential Approach	0.8917	0.8900	0.8040	52
New Algorithm	0.8692	0.9000	0.8001	103

computational time will tremendously increase in that case. Also in the next a few examples, however, sequential approach fails to provide a desired design.

Example II E_{max} model is commonly used in dose-finding studies. This model can be written as

$$y = \beta_0 + \frac{\beta_1 x}{\beta_2 + x} + \epsilon, \tag{15}$$

where x represents the dose level, ϵ is assumed to follow the normal distribution with mean zero and variance σ^2, β_0 represents the response when the dose level is at 0, $\beta_1(E_{max})$ is the maximum effect of the drug and $\beta_2(ED_{50})$ can be regarded as the dose level which produces half of E_{max}. In a dose finding study, (Dette, Bretz, Pepelyshev and Pinheiro, 2008) used Model (15) to find optimal design for the minimum effective dose level (MED) under parameter estimates $\beta_0 = 0$, $\beta_1 = 0.4760$, and $\beta_2 = 25$, where the relevant difference Δ is set as 0.2. Suppose a researcher is interested in estimating $h_0(\beta) = \beta_2$, $h_1(\beta) = \beta_1$, and $h_2(\beta) = MED = \beta_2 log(\frac{\beta_1+\Delta}{\beta_1})$. Let $c_i = \frac{\partial h_i(\beta)}{\partial \beta}$ and $\xi_i^* = argmin_\xi tr(c_i^T I^{-1}(\xi) c_i)$, $i = 0, 1, 2$.

Table 4: Example II: relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1.0000	0.5891	0.6670
ξ_1^*	0.0001	1.0000	0.0001
ξ_2^*	0.0028	0.0006	1.0000
ξ^*	0.9609	0.7008	0.6505

The corresponding objective functions can be written as

$$\Phi_0(I(\xi)) = -\frac{tr(c_0^T I^{-1}(\xi)c_0)}{tr(c_0^T I^{-1}(\xi_0^*)c_0)}, \text{ and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(c_i^T I^{-1}(\xi)c_i)}{tr(c_i^T I^{-1}(\xi_i^*)c_i)}, i = 1, 2.$$

Consider three-objective optimal design problem

$$\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)}$$
$$\text{subject to} \quad \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.7, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.65. \end{cases}$$

Utilizing the new algorithm, we find that the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.4944\Phi_1 + 0.2258\Phi_2.$$

The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and the constrained optimal design ξ^* under each of different optimal criteria are shown in Table 4.

Table 5 shows the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm. The table shows that the new algorithm produces a desired design. Grid search also produces a satisfied solution, although the computational time is around fifteen times of that of the new algorithm. A notable fact is that the sequential approach could not produce a proper solution. For sequential approach, all possible orders are tested and they all fail to produce a proper design. Sequential approach results based on different orders are shown in Table 6. ξ_{ijk}^* is the derived design based on the order $\Phi_i \to \Phi_j \to \Phi_k$ using the sequential approach. Since by the sequential approach procedure, ξ_{012}^* will be equivalent to ξ_{102}^* and ξ_{021}^* is equivalent to ξ_{201}^*, only four different orders are shown on the table. From Table 6, we can see that ξ_{210}^* performs relative good. However the efficiency for Φ_2 for ξ_{210}^* is 0.6726 while the corresponding constraint value is 0.65. This indicates ξ_{210}^* does not identify the

Table 5: Example II: relative efficiencies of constrained optimal designs based on different approaches

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Time Cost (Seconds)
Grid Search	0.9604	0.7000	0.6529	502
Sequential Approach		Failed		
New Algorithm	0.9609	0.7008	0.6505	34

Table 6: Example II: efficiencies of the derived designs based on different orders using sequential approach

	Efficiency		
Designs	Φ_0	Φ_1	Φ_2
ξ^*_{120}	0.9036	0.6992	0.6854
ξ^*_{210}	0.9437	0.6995	0.6726
ξ^*_{102}		Fails	
ξ^*_{201}		Fails	

active objective function Φ_2.

Example III (Atkinson, Chaloner, Juritz and Herzberg , 1993) derived Bayesian designs for a compartmental model which can be written as

$$y = \theta_3(e^{-\theta_1 x} - e^{-\theta_2 x}) + \epsilon = \eta(x, \theta) + \epsilon. \tag{16}$$

where ϵ is assumed to follow the normal distribution with mean zero and variance σ^2 and y represents the concentration level of the drug at time point x. (Clyde and Chaloner, 1996) derived multiple objective optimal designs under this model with parameter values $\theta^T = (\theta_1, \theta_2, \theta_3) = (0.05884, 4.298, 21.80)$ and design space $[0, 30]$. Interests are on estimating θ as well as the following quantities:

- Area under the curve (AUC),
$$h_1(\theta) = \frac{\theta_3}{\theta_1} - \frac{\theta_3}{\theta_2}$$
- Maximum concentration,
$$c_m = h_2(\theta) = \eta(t_{max}, \theta),$$
where $t_{max} = 1.01$.

Let $\xi_0^* = argmin|I^{-1}(\xi)|$, c_i be the gradient vector of $h_i(\theta)$ according to parameter vector θ and $\xi_i^* = argmintr(c_i^T I^{-1}(\xi) c_i)$, $i = 1, 2$. The corresponding objective functions can be

Table 7: Example III: relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1.0000	0.3431	0.3634
ξ_1^*	0.0036	1.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000
ξ^*	0.9761	0.4008	0.4046

written as follows:

$$\Phi_0(I(\xi)) = -(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_0^*)|})^{\frac{1}{3}}, \text{ and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(c_i^T I^{-1}(\xi)c_i)}{tr(c_i^T I^{-1}(\xi_i^*)c_i)}, i = 1, 2.$$

Consider the following three-objective optimal design problem:

$$\begin{array}{ll} \underset{\xi}{\text{Maximize}} & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} & \text{Effi}_{\Phi_i(\xi)} \geq 0.4, i = 1, 2. \end{array}$$

Utilizing the new algorithm, we find that the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0916\Phi_1 + 0.0854\Phi_2.$$

The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and the constrained optimal design ξ^* under different optimality criteria are shown in Table 7.

Table 8 shows the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm. The table clearly shows both new algorithm and grid search produce a satisfied solution. But grid search takes around eighteen times calculational time of that of the new algorithm. On the other hand, the sequential approach again fails to produce a satisfied solution. For sequential approach, all possible orders are tested and results are shown in Table 9. ξ_{ijk}^* is the sequential optimal design based on order $\Phi_i \to \Phi_j \to \Phi_k$. Table 9 shows sequential approach with order $\Phi_1 \to \Phi_0 \to \Phi_2$ and order $\Phi_2 \to \Phi_0 \to \Phi_1$ fails to produce a design which satisfies all the constraints. For optimal designs derived with the other two orders, although constraints are satisfied. The efficiency of the target objective function Φ_0 is far below the results from the new algorithm and the grid search. All these indicate that sequential approach may not be proper for finding multiple-objective optimal design problems.

Table 8: Example III: relative efficiency of constrained optimal design based on different techniques

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Time Cost (Seconds)
Grid Search	0.9761	0.4042	0.4009	1047
Sequential Approach		Fails		
New Algorithm	0.9761	0.4008	0.4046	59

Table 9: Example III: efficiencies of the derived designs based on different orders using sequential approach

	Efficiency		
Designs	Φ_0	Φ_1	Φ_2
ξ^*_{120}	0.5797	0.3908	0.5981
ξ^*_{210}	0.4537	0.6135	0.3904
ξ^*_{102}		Fails	
ξ^*_{201}		Fails	

5.2 Four-objective and Five-objective Optimal Designs

In this subsection, we shall mainly focus on the performance of the new algorithm when there are four or five objectives. The sequential approach is dropped due to its unstable performance. The grid search is not considered either due to its lengthy computational time.

Example IV Under the same set up as that of Example III, another parameter of interest, time to maximum concentration t_m is also considered, where

$$t_m = h_3(\theta) = \frac{log(\theta_2) - log(\theta_1)}{\theta_2 - \theta_1}.$$

The corresponding objective function is

$$\Phi_3(I(\xi)) = -\frac{tr(c_3^T I^{-1}(\xi)c_3)}{tr(c_3^T I^{-1}(\xi_3^*)c_3},$$

where c_3 is the gradient vector of $h_3(\theta)$ according to vector θ and $\xi_3^* = argmin_\xi tr(c_3^T I^{-1}(\xi)c_3)$. (Clyde and Chaloner, 1996) studied the following four-objective optimal design problem

$$\begin{aligned} \underset{\xi}{\text{Maximize}} \quad & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} \quad & \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.4, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.4, \\ \text{Effi}_{\Phi_3(\xi)} \geq 0.4. \end{cases} \end{aligned}$$

is considered.

Utilizing the new algorithm, we find that the corresponding Lagrange funcion is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0916\Phi_1 + 0.0854\Phi_2.$$

This indicates that only two out of the three constrains are active, which are objective functions Φ_1 and Φ_2. The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, and the constrained optimal design ξ^* under different optimal criteria are shown in Table 10. The computational time is around 56 seconds.

Example V Based on the same settings as Example IV, we add one more objective function:

$$\Phi_4(I(\xi)) = -\frac{tr(I^{-1}(\xi))}{tr(I^{-1}(\xi_4^*))}.$$

Here $\xi_4^* = argmintr(I^{-1}(\xi))$. Then five-objective optimal design problem

$$\begin{aligned} \underset{\xi}{\text{Maximize}} \quad & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} \quad & \begin{cases} \text{Effi}_{\Phi_i(\xi)} \geq 0.4, i = 1, 2, 3 \\ \\ \text{Effi}_{\Phi_4(\xi)} \geq 0.75 \end{cases} \end{aligned}$$

Table 10: Example IV: the relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^* and ξ^*

	Efficiency			
Design Type	Φ_0	Φ_1	Φ_2	Φ_3
ξ_0^*	1.0000	0.3431	0.3634	0.6464
ξ_1^*	0.0036	1.0000	0.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000	0.0002
ξ_3^*	0.0785	0.0001	0.0007	1.0000
ξ^*	0.9761	0.4008	0.4046	0.5143

Table 11: Example V: the relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and ξ^*

	Efficiency				
Design Type	Φ_0	Φ_1	Φ_2	Φ_3	Φ_4
ξ_0^*	1.0000	0.3431	0.3634	0.6464	0.7044
ξ_1^*	0.0036	1.0000	0.0000	0.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000	0.0002	0.0005
ξ_3^*	0.0785	0.0001	0.0007	1.0000	0.0010
ξ_4^*	0.7904	0.1138	0.6460	0.5895	1.0000
ξ^*	0.9616	0.4013	0.4184	0.4945	0.7501

is considered.

Result from new algorithm indicates that the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.3052\Phi_1 + 0.8362\Phi_4.$$

Only objective functions Φ_1 and Φ_4 are active in this case. The efficiencies of of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and the constrained optimal design ξ^* under different optimal criteria are shown in Table 11. It takes 2 minutes and 27 seconds for the new algorithm to find ξ^*.

Example VI

Consider Model (14) in Example I. Suppose that we want to maximize the efficiency of D-optimal while guarantee that the efficiency of C-optimal for each parameter is above 0.7. All other settings are as the same as those of example I. Let $\xi_0^* = argmin|I^{-1}(\xi)|$ and $\xi_i^* = argmintr(e_i^T I^{-1}(\xi) e_i)$, $i = 1, 2, 3, 4$, where e_i is the unit vector with i-th element equal to 1.

Table 12: Example VI: the relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and ξ^*

	Efficiency				
Design Type	Φ_0	Φ_1	Φ_2	Φ_3	Φ_4
ξ_0^*	1.0000	0.8323	0.4461	0.6326	0.5967
ξ_1^*	0.9141	1.0000	0.3294	0.6234	0.6136
ξ_2^*	0.3849	0.1964	1.0000	0.3353	0.6422
ξ_3^*	0.1471	0.0006	0.0232	1.0000	0.0051
ξ_4^*	0.6044	0.4260	0.6867	0.6230	1.0000
ξ^*	0.9259	0.7009	0.7007	0.7212	0.7027

The corresponding objective functions can be written as following:

$$\Phi_0(I(\xi)) = -(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_0^*)|})^{\frac{1}{3}}, \text{and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(e_i^T I^{-1}(\xi)e_i)}{tr(e_i^T I^{-1}(\xi_i^*)e_i)}, i = 1, 2, 3, 4.$$

Consider the following five-objective optimal design problem

$$\begin{aligned} \underset{\xi}{\text{Maximize}} \quad & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} \quad & \text{Effi}_{\Phi_i(\xi)} \geq 0.7, i = 1, 2, 3, 4. \end{aligned}$$

Results from new algorithm show that the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0183\Phi_1 + 0.3540\Phi_2 + 0.0305\Phi_4.$$

Only objective function Φ_3 is inactive in this case. The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and the constrained optimal design ξ^* under different optimal criteria are shown in Table 12. It takes about 42 minutes on a laptop.

6 Discussion

While the importance of multiple objective optimal designs is well recognized in scientific studies, their applications are still undeveloped due to a lack of a general and efficient algorithm. The combination of OWEA algorithm for compound optimal design problem and the new algorithm provides an efficient and stable framework for finding the general multiple-objective optimal designs. Examples show that we can easily find the optimal designs for multiple-objective design problems with a laptop even when there are more than four objective functions involved. The difference of computational cost between the grid search and the new algorithm become more and more remarkable as increase of required accuracy.

In the process of solving multiple-objective optimal design problems, the new algorithm searches for the corresponding compound optimal design problem from the simplest case - all constraints are inactive to the most complex case - all constraints are active. During the searching process, once a weight vector U^* satisfying the sufficient condition is found, the new algorithm stops and outputs that U^*. The corresponding compound optimal design can be constructed based on U^*.

For optimal designs with no more than four objective functions, the new algorithm can derive the desired solution efficiently. When there are five or more objective functions, it is unlikely all constraints are active. If only less than four constraints are active, the new algorithm can still solve the optimal design efficiently. However, in a rare situation where there are four or more active constraints, the computation time can become lengthy. More research works are needed to deal with these cases.

To guarantee the convergence of new algorithm, the strict concavity of the objective function Φ_0 is required. However, various cases are tested and the convergence hold for virtually all situations based on our experience. Like in Example II, Φ_0 is just a concave function, but the new algorithm still performs properly. It may be worthwhile to look into the theoretical properties for these cases. On the other hand, the new algorithm is implemented under locally optimal designs context for all examples. It is possible to extend the results to other settings, like to the cases dicussed in (Cook and Fedorov, 1995). More research works are certainly needed to realize this idea.

Although computer codes of this new algorithm is not straightforward, the main body of the code work for all multiple-objective design problems. One only needs to change the information matrix for the specific model and the specific objective functions in a multiple-objective optimal design problem. The SAS IML codes for all examples in this article can be downloaded from http://homepages.math.uic.edu/∼minyang. These codes can be easily modified for different multiple objective optimal problems.

7 Appendix

Let $S \subset \{1, \dots, n\}$, for easy presentation, we denote $\mathbf{U}_S \hat{\Phi}_S(\xi) = \sum_{i \in S} u_i \hat{\Phi}_i(\xi)$. We also denote $\hat{\Phi}(\xi) = (\hat{\Phi}_1(\xi), \dots, \hat{\Phi}_n(\xi))$.

Proof of Theorem 2. Let $u_a^0 > u_a^1$ be two nonnegative values. Let $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$ be the corresponding value sets for $\mathbf{U}_S$ satisfying the two conditions in the theorem when $u_a = u_a^0$ and u_a^1, respectively. Let $\mathbf{U}^0$ be the combination of $\mathbf{U}_S^0$, u_a^0, and $\mathbf{U}_{S'}$ by their corresponding indexes. Similarly let $\mathbf{U}^1$ be the counterpart of $\mathbf{U}_S^1$, u_a^1, and $\mathbf{U}_{S'}$.

Notice that for $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$, the classification of S_1 and S_2 could be different. That means elements in S_1 for $\mathbf{U}_S^0$ may fall into S_2 for $\mathbf{U}_S^1$ and versus the same. We just need to check that the two disjoint subsets from S satisfy Condition (8) in the theorem separately.

By the properties of $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$, we have

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^1}), \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^1}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^0}). \end{aligned} \tag{17}$$

Notice that

$$
\begin{aligned}
(\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^0}) &= (\mathbf{U}_S^0)^T\hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u_a^0\hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^T\hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}),\\
(\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^1}) &= (\mathbf{U}_S^0)^T\hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u_a^0\hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^T\hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}),\\
(\mathbf{U}^1)^T\hat{\Phi}(\xi_{\mathbf{U}^0}) &= (\mathbf{U}_S^1)^T\hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u_a^1\hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^T\hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}), \text{ and}\\
(\mathbf{U}^1)^T\hat{\Phi}(\xi_{\mathbf{U}^1}) &= (\mathbf{U}_S^1)^T\hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u_a^1\hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^T\hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}).
\end{aligned}
\tag{18}
$$

Adding up the two inequalities in (17) and utilizing (18), we have

$$
(u_a^0 - u_a^1)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) + (\mathbf{U}_S^0 - \mathbf{U}_S^1)^T(\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) \geq 0. \tag{19}
$$

Suppose $i \in S_1$ when $u_a = u_a^0$ and $i \in S_2$ when $u_a = u_a^1$. Clearly that $(u_i^0 - u_i^1) \leq 0$ while $(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \geq 0$. The conclusion holds for all other cases through the similar argument. Thus we have, for any $i \in S$, $(u_i^0 - u_i^1)(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \leq 0$. Consequently, we have

$$
(\mathbf{U}_S^0 - \mathbf{U}_S^1)^T(\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) = \sum_{i\in S}(u_i^0 - u_i^1)(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \leq 0, \tag{20}
$$

which indicates

$$
(u_a^0 - u_a^1)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) \geq 0. \tag{21}
$$

Thus the conclusion follows. ■

Proof of Theorem 3. By the definitions of $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$, we have

$$
\begin{aligned}
\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^0}) &\geq \Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^1}), \text{ and}\\
\Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^1)^T\hat{\Phi}(\xi_{\mathbf{U}^1}) &\geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^1)^T\hat{\Phi}(\xi_{\mathbf{U}^0}).
\end{aligned}
\tag{22}
$$

By (9), (22) can be rewritten as

$$
\begin{aligned}
\Phi_0(\xi_{\mathbf{U}^0}) &\geq \Phi_0(\xi_{\mathbf{U}^1}), \text{ and}\\
\Phi_0(\xi_{\mathbf{U}^1}) &\geq \Phi_0(\xi_{\mathbf{U}^0}),
\end{aligned}
\tag{23}
$$

which implies

$$
\Phi_0(\xi_{U^0}) = \Phi_0(\xi_{U^1}). \tag{24}
$$

Since Φ_0 is strictly concave function on information matrices, $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$ have the same information matrix. Thus $\xi_{\mathbf{U}^0}$ is equivalent to $\xi_{\mathbf{U}^1}$.

■

Proof of Theorem 4. Define $S_{11} = \{i|\hat{\Phi}_i(\xi_{\mathbf{U}^0}) > 0, i \in S_1\}$. By the properties of $\mathbf{U}^0$, clearly we have $u_i^0 = 0$ for $i \in S_{11}$ and $u_i^0 = N_i$ for $i \in S_2$. Suppose there exists a positive value set $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$ with $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for $i \in S$. Then we have

$$\begin{aligned}&\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^0)^T\hat{\Phi}(\xi_{\mathbf{U}^+}) \text{ and}\\ &\Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^+)^T\hat{\Phi}(\xi_{\mathbf{U}^+}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^+)^T\hat{\Phi}(\xi_{\mathbf{U}^0}).\end{aligned} \tag{25}$$

Then the summation of two inequalities in (25) returns

$$\begin{aligned}&(\mathbf{U}^0_{S/(S_{11}\cup S_2)} - \mathbf{U}^+_{S/(S_{11}\cup S_2)})^T(\hat{\Phi}_{S/(S_{11}\cup S_2)}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S/(S_{11}\cup S_2)}(\xi_{\mathbf{U}^+}))\\ &+ (\mathbf{U}^0_{S_{11}} - \mathbf{U}^+_{S_{11}})^T(\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^+})) + (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S_2}(\xi_{\mathbf{U}^+})) \geq 0.\end{aligned} \tag{26}$$

By Condition (10) and our assumption, we have (i) $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0$ for $i \in S/(S_{11} \cup S_2)$; (ii) $u_i^0 = 0$ for $i \in S_{11}$; and (iii) $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for $i \in S$. Thus, (26) can be reduced to

$$(-\mathbf{U}^+_{S_{11}})^T\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})) \geq 0. \tag{27}$$

Notice that, for $i \in S_{11}$, $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) > 0$ and $\mathbf{U}^+_{S_{11}} > 0$. We have

$$(-\mathbf{U}^+_{S_{11}})^T\hat{\Phi}_{S_{11}}(\xi_{U^0}) < 0. \tag{28}$$

On the other hand, for $i \in S_2$, $u_i^0 = N_i > \mathbf{U}_i^+$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0$, we have

$$(U^0_{S_2} - U^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{U^0})) < 0. \tag{29}$$

Since $S_{11} \bigcup S_2 \neq \emptyset$, we have

$$(-\mathbf{U}^+_{S_{11}})^T\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})) < 0. \tag{30}$$

This is contradiction to (27). Thus the conclusion follows. ■

References

Atkinson, A. C., Chaloner, K., Juritz, J. and Herzberg, A. M. (1993). Optimum experimental designs for properties of a compartmental model. *Biometrics*, 49:325–337.

Chernoff, H. (1953). Locally optimal designs for estimating parameters. *Annals of Mathematical Statistics*, 24:586–602.

Clyde, M. and Chaloner, K.(1996). The equivalence of constrained and weighted designs in multiple objective design problems. *Journal of the American Statistical Association*, 91:1236–1244.

Cook, D. and Fedorov, V. (1995). Constrained optimization of experimental design. *Statistics*, 26:129–178.

Cook, R. D. and Wong W. K. (1994). On the equivalence of constrained and compounded optimal designs. *Journal of the American Statistical Association*, 89:687–692.

Dette, H., Bretz, F., Pepelyshev, A. and Pinheiro, J. (2008). Optimal designs for dose-finding studies. *Journal of the American Statistical Association*, 103:1225–1237.

Dette, H., Melas, V. B. (2011). A note on the de la garza phenomenon for locally optimal designs. *The Annals of Statistics*, 39:1266–1281.

Dette, H., Schorning, K. (2013). Complete classes of designs for nonlinear regression models and principal representations of moment spaces. *The Annals of Statistics*, 41:1260–1267.

Huang, Y. C. and Wong W. K. (1998). Sequential construction of multiple-objective optimal designs. *Biometrics*, 54:1388–1397.

Huang, Y. C. (1996). Multiple objective optimal designs. Dr.P.H. dissertation, School of Public Health, Department of Biostatistics, University of California at Los Angeles.

Notari, R. E. (1980). Biopharmaceutics and Clinical Pharmacokinetics. Introduction. New York and Basel: Marcel Dekker

Rosenberger, W. F. and Gril, S. E. (1997). A sequential design for psychophysical experiments: an application to estimating timing of sensory events. *Statistics in Medicine*, 16:2245–2260.

Yang, M. (2010). On the de la garza phenomenon. *The Annals of Statistics*, 38:2499–2524.

Yang, M., Biedermann, S. and Tang, E. (2013). On optimal designs for nonlinear models: a general and efficient algorithm. *Journal of the American Statistical Association*, 108:1411–1420.

Yang, M., and Stufken, J. (2009). Support points of locally optimal designs for nonlinear models with two parameters . *The Annals of Statistics*, 37:518–541.

Yang, M., and Stufken, J. (2012). Identifying locally optimal designs for nonlinear models: A simple extension with profound consequences. *The Annals of Statistics*, 40:1665–1681.