Journal of Statistical Planning and Inference 200 (2019) 87–101

Contents lists available at ScienceDirect

Journal of Statistical Planning and Inference

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

On multiple-objective optimal designs

Qianshun Cheng, Min Yang *

University of Illinois at Chicago, United States

ARTICLE INFO

Article history:
Received 21 February 2018
Received in revised form 30 August 2018
Accepted 5 September 2018
Available online 22 September 2018

Keywords:
Constrained designs
Compound designs
Algorithm

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. 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

Experiments with multiple objectives form a staple diet of modern scientific research. For example, 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}. A design that is efficient for the estimation of one of these dose levels is unlikely to be efficient for the others. In an example with four objectives, Clyde and Chaloner (1996) showed that an optimal design for one of the objectives reached efficiencies of only 7%, 10%, and 39% for the others.

There are several approaches for finding efficient designs when multiple objective functions are considered. One approach, which is popular in the optimization literature, is to establish a Pareto front or boundary. Pareto front approach aims to account for all criteria simultaneously by developing a set of Pareto optimal designs and investigate the trade-offs between different designs. For example, Kao et al. (2012) used a modified non-dominated sorting genetic algorithm to obtain a Pareto boundary in the context of event-related fMRI experiments; Cao et al. (2015) proposed a framework for comparing algorithm-generated Pareto fronts based on a refined hypervolume indicator. The graphical presentations they used are effective for two or three criteria, but may not for larger number of criteria. In this paper, we will not pursue this approach any further.

A second approach would use a compound optimality criterion that is a weighted sum of the individual objective functions. An attractive feature is that, for given weights, the compound criterion maintains the concavity property if the separate objective functions possess this property. This property is critically important for applying the celebrated equivalence theorem, which enables verification whether a given design is indeed optimal. With this approach, the weight assigned to each objective function is pre-specified. Then the design found is optimal according to the newly constructed weighted objective function. However, the choice of weights is the main difficulty with this approach; it does in general not have a meaningful interpretation.

The third approach is the constrained optimization approach. It formulates the optimality problem as maximizing one objective function subject to all other objective functions satisfying certain efficiencies. The constrained optimization approach provides a clear and intuitive interpretation to the multiple objective design problem, making it become one of the popular approaches for finding multiple objective optimal design.

* Corresponding author.
E-mail addresses: chengqianshun1@gmail.com (Q. Cheng), myang2@uic.edu (M. Yang).

https://doi.org/10.1016/j.jspi.2018.09.007
0378-3758/

On the flip side, in contrast to the compound optimality approach, with the constrained approach there is no "equivalence theorem" that allows a user to verify whether a solution is indeed optimal. 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 criterion. 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 using this relation: 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 170 000. Since the best design must be found for each of these, the grid search will become very quickly computationally infeasible as the accuracy increases. And 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 satisfactory solution even for the three-objective optimal design problems.

Other approaches are also available. Mikulecka (1983) proposed the idea of hybrid design and algorithm to numerically find the optimal design based on hybrid design settings, which can be regarded as trying to optimize the compound optimal design problem while meeting one constraint criteria. Vandenberghe et al. (1998) proposed an interior-point method to solve determinant maximization problem with linear matrix inequality constraints, which can be used to solve some of the constrained optimal design problem. Harmon and Benkova (2017) proposed the Barycentric algorithm specific for computing D-optimal size- and cost-constrained designs of experiments. Mandal et al. (2005) considered constructing constrained optimal designs with equality constraints and Sagnol and Harman (2015) focused on finding optimal designs with system of linear constraints on weight vectors of design points. The approach proposed by Sagnol and Harman (2015) theoretically can solve the proposed problem if the primary and secondary criteria meet certain requirements. However, technically it is very challenging to derive a practical algorithm based on that. Thus they are not discussed here.

The goal of this paper is to propose a novel algorithm of deriving the optimal design of a given constrained optimality problem through finding the weights in the corresponding compound design. Consistency of the algorithm is proved. The performance of the new algorithm is demonstrated by comparing with the grid search approaches and sequential approaches. In 2016, a short version of this work (Cheng et al., 2016) is presented and published in the 11th International Workshop in Model-Oriented Design and Analysis. In that version, we skipped all the proofs and a few examples. And this paper includes all the proof details and more examples.

This paper is organized as follows. In Section 2, we introduce the set up and necessary notation. Characterization and convergence properties are presented in Section 3. The implementation of the algorithm, as well as the computational cost discussion is in Section 4. Applications to three examples with different number of constraints, and comparisons with grid search and sequential approach are shown in Section 5. Section 6 provides a brief discussion. For the space limit, we put all the proofs and some of the examples in the Appendix.

2. Set up and notation

We adapt the same notation as those of Yang et al. (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 $\eta(\mathbf{x},\theta)$, where θ is a $(k \times 1)$ vector of unknown parameters. Typically, approximate designs are studied, i.e. designs of the form $\xi = \{(\mathbf{x}_i, \omega_i), i = 1, \dots, m\}$ 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 $\Phi_0(\xi), \dots, \Phi_n(\xi)$ 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 objectives may have different parameters of interest. For example, $\Phi_0(\xi)$ can be the negative number of the trace of inverse of the information matrix; $\Phi_1(\xi)$ can be the negative 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 $\Phi_0(\xi), \dots, \Phi_n(\xi)$ simultaneously among all possible designs. However, such solution does not exist in general. Constrained optimization approach specifies one objective as the primary criteria and maximizes this objective subject to the constraints defined based on the remaining objectives (Cook and Wong, 1994; Clyde and Chaloner, 1996). Formally, this approach can be written as

$$\underset{\xi \in \Xi}{\text{Maximize}}\ \Phi_0(\xi) \text{ subject to } \Phi_i(\xi) \geq c_i,\ i = 1, \dots, n, \tag{2.1}$$

where $\mathbf{c} = (c_1, \dots, c_n)$ 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.

Unfortunately, with the restricted optimality set up, there is a lack of direct and computational feasible way to generally solve the constrained optimization problem, especially when we have many constraint criteria. (2.1) can be solved typically through the corresponding compound optimal design. Let

$$L(\xi, \mathbf{U}) = \Phi_0(\xi) + \sum_{i=1}^{n} u_i(\Phi_i(\xi) - c_i), \tag{2.2}$$

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. Once a $\mathbf{U}$ is given, deriving a design maximizing $L(\xi, \mathbf{U})$ can be based on some existing algorithms, such as PSO (Mandal et al., 2015); Cocktail algorithm (Yu, 2011); and OWEA (Yang et al., 2013), among others. As we mentioned before, it is not recommended to use compound optimal design strategy directly due to lack of 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 in (2.1) 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 2.1 (*Clyde and Chaloner, 1996*)**.** *Under assumptions A1 to A4, ξ^* is optimal for constrained optimal design (2.1) if and only if there exists a non-negative vector $\mathbf{U}^* = (u_1^*, \ldots, u_n^*) \in \Re^n$, such that*

$$\xi^* = \underset{\xi \in \Xi}{\text{argmax}}\, L(\xi, \mathbf{U}^*),\ \Phi_i(\xi^*) \geq c_i \text{ for } i = 1, \ldots, n$$
$$\text{and } \sum_{i=1}^{n} u_i^*(\Phi_i(\xi^*) - c_i) = 0. \tag{2.3}$$

Theorem 2.1 provides necessary and sufficient condition for constrained optimal designs (2.1). It demonstrates that a numerical solution for the constrained design problem (2.1) can be derived by using an appropriate compound optimality criterion. The big challenge is how to find the desired $\mathbf{U}^*$ for a given constrained design problem (2.1). Since the explicit forms of the derivatives are not available, direct use of derivative based algorithms to find this $\mathbf{U}^*$ may not be accurate and may lead to some undesired local roots. Thus they are not discussed here. There are two approaches to handle this: the grid search approach and the sequential approach. Both approaches consider the weighted optimal design, which is equivalent to compound optimal design. Let

$$\Phi_\lambda(\xi) = \sum_{i=0}^{n} \lambda_i \Phi_i(\xi), \tag{2.4}$$

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 (2.1). 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 one assumption. Let ξ^* be the optimal design for a constrained optimal design problem (2.1). By Theorem 1, ξ^* is also an optimality solution of a compound optimal design problem (2.2). Let $\mathbf{U}^* = (u_1^*, \ldots, u_n^*)$ be the Lagrange multiplier of the compound optimal design problem.

In such a compound optimal design problem (2.2), each $u_i > 0$ without upper bound. However, for an algorithm searching for $\mathbf{U}^*$, it is challenging to establish the convergence property of the algorithm when the search space is infinite. Thus our assumption is

$$u_i^* \in [0, N_i) \text{ where } N_i \text{ is pre-specified}, i = 1, \ldots, n. \tag{3.1}$$

This assumption is equivalent to the grid size in a weighted optimal design problem (2.4). Both grid search approach and sequential approach need to choose a grid size. Let the grid size be ϵ, then it means $0 \le u_i \le \frac{1-\epsilon}{\epsilon} < \frac{1}{\epsilon}$ for the equivalent compound optimal design (2.2). We can always choose some reasonable large numbers N_i's such that Assumption (3.1) 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 $\mathbf{U} = (u_1, \ldots, u_n)$ 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 will be given in Section 4.

3.1. Overview of the new algorithm

The new algorithm is designed to search for a satisfied $\mathbf{U}^*$ from the easiest case to the most complex case. It will go through all the possible cases following a complexity order until the right combination of active constraints is found:

$$\begin{aligned}&\text{All constraints are inactive} \longrightarrow \text{One constraint is active}\\ &\longrightarrow \cdots \longrightarrow \text{All constraints are active.}\end{aligned}$$

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

To search for satisfied values for $u_1, \ldots, u_a$, the algorithm will use bisection process for all elements $u_1, \ldots, u_a$ through an iterative procedure. The rest element $u_{a+1}, \ldots, u_n$ in weight vector $\mathbf{U}$ will be fixed at 0 during the bisection process. Denote the final weight function parameter $\mathbf{U}$ obtained from this bisection procedure by $\mathbf{U}^* = (u_1^*, \ldots, u_a^*, 0, \ldots, 0)$. Then for any $i \in \{1, \ldots, a\}$, u_i^* will satisfy the following property:

$$\begin{aligned}&\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) > 0, \text{ then } u_i^* = 0;\\ &\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) < 0, \text{ then } u_i^* = N_i;\\ &\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0, \text{ then } u_i^* \in [0, N_i].\end{aligned} \tag{3.2}$$

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 (3.2) with $\mathbf{U}^0 = (u_1^0, u_2^0, 0, \ldots, 0)$. Now check $\hat{\Phi}_2(\xi_{U^0})$. If $\hat{\Phi}_2(\xi_{\mathbf{U}^0}) \ne 0$, 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 (3.2) with $U^1 = (u_1^1, u_2^1, 0, \ldots, 0)$. Check $\hat{\Phi}_2(\xi_{U^1})$ and update u_2 to u_2^2 if $\hat{\Phi}_2(\xi_{U^1}) \ne 0$. Continue this process until a satisfied $\mathbf{U}^* = (u_1^*, u_2^*, 0, \ldots, 0)$ is found which guarantees that u_1^* and u_2^* both satisfy Property (3.2).

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 recursive procedure mentioned for 2 active constraints case, derive the corresponding values $u_1^0, \ldots, u_{a-1}^0$ for the element u_1 to u_{a-1} using bisections approach such that they satisfy Property (3.2) with $\mathbf{U}^0 = \{u_1^0, \ldots, u_a^0, 0, \ldots, 0\}$. Check whether $\hat{\Phi}_a(\xi_{\mathbf{U}^0}) = 0$ and update u_a to u_a^1. Continue this process until a desired $\mathbf{U}^* = (u_1^*, \ldots, u_a^*, 0, \ldots, 0)$ is found with all $u_1^*, \ldots, u_a^*$ that satisfied Property (3.2).

To guarantee the bisection technique is valid and the desired Property (3.2) can be achieved for $u_1, \ldots, u_a$ through the bisection process, we need to characterize the monotone 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. Theorems

Theorem 3.1. *For any $a \in \{1, \ldots, n\}$, $S \subsetneq \{1, \ldots, n\} \setminus \{a\}$ and $S' = \{1, \ldots, n\} \setminus (S \bigcup \{a\})$, 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:*

$$\begin{aligned}&\hat{\Phi}_i(\xi_{\mathbf{U}}) \ge 0 \text{ and } u_i \hat{\Phi}_i(\xi_{\mathbf{U}}) = 0 \text{ for } i \in S_1, \text{ or}\\ &u_i = N_i \text{ and } \Phi_i(\xi_{\mathbf{U}}) < 0 \text{ for } i \in S_2,\end{aligned} \tag{3.3}$$

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.

The main purpose of Theorem 3.1 is to guarantee that the recursive bisection technique can be properly implemented. Condition (3.3) implies that u_i, $i \in S$ satisfy Property (3.2). Suppose there are a active constraints and they are $\Phi_1, \ldots, \Phi_a$. When we search for the proper value of u_i $(i \le a - 1)$, $u_{i+1}, \ldots, u_a$ and the zero-element $u_{a+1}, \ldots, u_n$ can be regarded as fixed, which corresponds to $\mathbf{U}_{S'}$ in theorem. And since it is a recursive procedure, for $u_1, \ldots, u_{i-1}$, the value will be updated first according to the value assigned to u_i on each bisection iteration and fixed $u_{i+1}, \ldots, u_n$. Thus $(u_1, \ldots, u_{i-1})$ is $\mathbf{U}_S$ in this case. After $u_1, \ldots, u_{i-1}$ being updated for the given u_i, $\hat{\Phi}_i(\xi_{\mathbf{U}})$ should be a monotone increasing function of u_i by Theorem 3.1. 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 (3.2). Under all these possible cases that may occur when the bisection technique is applied to the former elements, Theorem 3.1 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_S with $S \subseteq \{1, \ldots, n\}$. A weight vector $\mathbf{U}_S^*$ for active constraints can be found through the bisection technique. One can always construct a complete weight vector $\mathbf{U}^* = (u_1^*, \ldots, u_n^*)$ as follows:

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

- 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 3.2. *For any $S \subset \{1, \ldots, n\}$, suppose that $\mathbf{U}^0 = \{\mathbf{U}_S^0, 0\}$ satisfies the following two conditions*

$$\begin{aligned} &(i)\ \hat{\Phi}_i(\xi_{\mathbf{U}^0}) \geq 0 \text{ for } i \in S_1 \text{ and } \sum_{i \in S_1} u_i \hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0. \\ &(ii)\ \hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0 \text{ and } u_i = N_i \text{ for } i \in S_2. \end{aligned} \tag{3.4}$$

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 $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0$, then there does not exist a non-negative value set $\mathbf{U}_S^+ = \{u_i \in [0, N_i) | i \in S\}$, such that $u_i \hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) \geq 0$ for $i \in S$, where $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$.

This theorem will help us prove the convergence of the new algorithm.

4. Algorithm

For a given constrained optimal design problem (2.1), 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 (2.2) with $\mathbf{U}$ being given. We first introduce such algorithm.

4.1. Deriving compound optimal design with given U

Yang et al. (2013) proposed the optimal weight exchange algorithm (OWEA), which 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. A detail description about OWEA algorithm can be found in the Appendix.

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). For each case, the algorithm will implement a recursive bisection procedure. The algorithm can be described as following:

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

Step 2 Set $i = 1$, consider $\xi^* = argmax_\xi\ \Phi_0(\xi) + u_i \Phi_i(\xi)$. 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 $\xi^* = argmax_\xi\ \Phi_0(\xi) + u_i^* \Phi_i(\xi)$, check whether $\hat{\Phi}_j(\xi^*) \geq 0$ for $j = 1, \ldots, n$. 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 $\{1, \ldots, n\}$ 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 recursive procedure. For each time a given value of u_{s_a}, first use bisection technique to find the corresponding $u_{s_1}, \ldots, u_{s_{a-1}}$. The full weight vector $\mathbf{U}$ can be constructed with $u_{s_1}, \ldots, u_{s_a}$ 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 $\{u^*_{s_1}, \ldots, u^*_{s_{a-1}}\}$ 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 (2.1).

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 vectors 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 4.1. *For the constrained optimal design problem (2.1), under Assumption (3.1), the proposed algorithm converges to* ξ^*.

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 (2.4), 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 $\lambda_1, \ldots, \lambda_n$ 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

$$T_G = \sum_{k=0}^{n} \binom{n}{k} \binom{\lfloor \frac{1}{\epsilon} \rfloor - 1}{k} = \binom{n + \lfloor \frac{1}{\epsilon} \rfloor - 1}{n}, \tag{4.1}$$

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 to the ceiling function. Let T_L be the number of times compound optimal designs calculated during the searching process, then

$$T_L = \sum_{k=0}^{n} \binom{n}{k} \lceil -2log_2\epsilon + 2 \rceil^k = \lceil -2log_2\epsilon + 3 \rceil^n. \tag{4.2}$$

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.

Table 1
Comparison of computational cost.

Mesh grid size	Three objectives		Four objectives	
	0.01	0.001	0.01	0.001
Grid search	5050	500 500	171 700	167 167 000
New algorithm	289	529	4 913	12 167

Note: Numbers in the table are counts of weighted optimal designs calculated to solve the multiple-objective design problem for each technique.

Table 1 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 reorders $\Phi_0, \ldots, \Phi_n$ as $\Phi_{s_1}, \ldots, \Phi_{s_{n+1}}$ 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). Since we do not have a real constraint number c_0 for target optimality Φ_0, here we can regard the constraint number c_0 for optimality Φ_0 as 0 and then combine c_0 with the original constraints vector $c = (c_1, \ldots, c_n)$. For newly constructed $c^* = (c_0, \ldots, c_n)$, reorder it as $(c_{s_1}, \ldots, c_{s_{n+1}})$. Details of this approach can be found in the Appendix.

For the grid search, weighted optimal design $\xi_\Lambda = \text{argmax}_\xi \sum_{i=0}^n \lambda_i \Phi_i(\xi)$ will be considered. All combinations of $\Lambda = (\lambda_0, \ldots, \lambda_n)^T$ 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 maximize Φ_0 while guaranteeing that $\Phi_i \geq c_i$ for $i = 1, \ldots, n$, 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 et al. (2013). For new algorithm and grid search, we require the algorithms to produce the best possible design while guaranteeing that the constraints are exactly satisfied. For sequential approach, since it does not 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 has $\Phi_i(\xi_0) \geq c_i - \epsilon$ 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.

Example I. Consider the nonlinear model given by

$$y = \beta_1 e^{-\theta_1 x} + \beta_2 e^{-\theta_2 x} + \epsilon. \tag{5.1}$$

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 (5.1) 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 $B = \text{diag}\{\frac{1}{\theta_1^2}, \frac{1}{\theta_2^2}, \frac{1}{\beta_1^2}, \frac{1}{\beta_2^2}\}$; $W = \int_2^{10} f(x)f^t(x)\upsilon(dx)$, 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 follows:

$$\Phi_0(I(\xi)) = -\frac{tr(I^{-1}(\xi)B)}{tr(I^{-1}(\xi_0^*)B)},$$

$$\Phi_1(I(\xi)) = -\left(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_1^*)|}\right)^{\frac{1}{4}}, \text{ and}$$

$$\Phi_2(I(\xi)) = -\frac{tr(I^{-1}(\xi)W)}{tr(I^{-1}(\xi_2^*)W)}.$$

Define $\text{Effi}_{\Phi_i(\xi)} = -\frac{1}{\Phi_i(I(\xi))}$. 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 to the D-efficiency. Such definition will be used in the subsequent examples.

Table 2
Example I: Relative design efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and ξ^*.

Design type	Efficiency		
	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 based on different techniques.

Techniques	Efficiency			Time cost (s)
	Φ_0	Φ_1	Φ_2	
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

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

$$\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)}$$

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

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 guaranteeing 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 slight 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 computational time will increase significantly in that case. In the next example, however, sequential approach fails to provide a desired design.

Example II. Atkinson et al. (1993) derived Bayesian designs for a compartmental model which can be written as

$$y = \theta_3(e^{-\theta_1 x} - e^{-\theta_2 x}) + \epsilon = \eta(x, \theta) + \epsilon. \tag{5.2}$$

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 $t_{max} = 1.01$.

Table 4
Example II: relative efficiencies of $\xi_0^*, \xi_1^*, \xi_2^*$, and ξ^*.

Design type	Efficiency		
	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

Table 5
Example II: relative design efficiency based on different approaches.

Techniques	Efficiency			Time cost (s)
	Φ_0	Φ_1	Φ_2	
Grid search	0.9761	0.4042	0.4009	1047
Sequential approach		Fails		
New algorithm	0.9761	0.4008	0.4046	59

Table 6
Example II: design efficiencies based on different orders using sequential approach.

Designs	Efficiency		
	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	

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 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:

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

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 $\xi_0^*, \xi_1^*, \xi_2^*$, and the constrained optimal design ξ^* under different optimality 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 clearly shows both new algorithm and grid search produce a satisfied solution. But grid search takes around eighteen times the computational 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 6. ξ_{ijk}^* is the sequential optimal design based on order $\Phi_i \to \Phi_j \to \Phi_k$. Table 6 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.

For the next example and the examples in the Appendix, the sequential approach is dropped due to its unstable performance and the grid search is not considered either due to its lengthy computational time.

Example III. Consider Model (5.1) in Example I. Suppose that we want to maximize the efficiency of D-optimal while guaranteeing 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 ith element equal to 1.

Table 7
Example III: the relative efficiencies of $\xi_0^*, \xi_1^*, \xi_2^*, \xi_3^*, \xi_4^*$, and ξ^*.

Design type	Efficiency				
	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

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

Results from the 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 $\xi_0^*, \xi_1^*, \xi_2^*, \xi_3^*, \xi_4^*$ and the constrained optimal design ξ^* under different optimal criteria are shown in Table 7. It takes about 37 min on a laptop.

6. Discussion

While the importance of multiple-objective optimal designs is well recognized in scientific studies, applications to solve this type of problems 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 remarkable improvement on computational cost compare with the grid search approach.

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.

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 discussed in Cook and Fedorov (1995). Penalty approaches are another strategy for finding multiple-objective optimal design. When implementing penalty approach, each constraint will be transferred to a penalty term. Thus the constrained optimal design problem can be transferred to a compounded optimal design problem with these penalty terms as the new optimal criteria. However, it is out of the scope of this paper. More future research works are certainly needed to realize the idea of penalty approach.

Although the computer codes of this new algorithm are not straightforward, the main body of the code should 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 are freely available upon request. These codes can be easily modified for different multiple-objective optimal problems.

Acknowledgments

The authors are thankful for detailed comments and suggestions by the associate editor and one referee on an earlier version of the article, which clearly helped to improve the final version. Min Yang's research was supported by National Science Foundation grants DMS-1407518 and DMS-1811291.

Appendix

A.1. OWEA algorithm

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 criterion. For a design $\xi = \{(\mathbf{x}_1, w_1), \ldots, (\mathbf{x}_{m-1}, w_{m-1}), (\mathbf{x}_m, w_m)\}$, let $X = (\mathbf{x}_1, \ldots, \mathbf{x}_m)^T$ and $W = (w_1, \ldots, w_{m-1})^T$. The following algorithm follows the similar procedure as that of OWEA in Yang et al. (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 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 et al. (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 $\hat{\mathbf{x}}_t = argmax_{\mathbf{x} \in \mathcal{X}}\ d_{\mathbf{U}}(\mathbf{x}, \xi^t)$. 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 $L(\xi, \mathbf{U})$ with respect to the weight vector W. These derivatives can be derived using (A.1) and the formula in the Appendix of Yang et al. (2013).

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

Based on the exact same argument as Yang et al. (2013), this algorithm converges to an optimal design maximizing $L(\xi, \mathbf{U})$. We use the extended OWEA to derive $\xi_{\mathbf{U}}$.

A.2. Sequential approach procedures

Then the sequential procedure for finding the corresponding compound optimal design with the specified order $\{s_1, \ldots, s_{n+1}\}$ 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

  Maximize Φ_0 while $\Phi_{s_2} \geq c_{s_2}$.

  If not, consider constrained optimal design problem

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

  Then find the weight vector in the weighted optimal design problem corresponding to the specified constrained optimal design problem using the grid search with a prefixed grid size and denote this 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 $(1 - x)\Phi_{\{s_1,\ldots,s_{k-1}\}} + x\Phi_{s_k}$. Change the value of x by grid search on [0, 1] with given grid size. If $\Phi_0 \in \{\Phi_{s_1}, \ldots, \Phi_{s_k}\}$, choose a proper value x such that the corresponding weight design maximizes Φ_0 while guarantees $\Phi_{s_i} \geq c_i$ for $i = 1, \ldots, k$. 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 $i = 1, \ldots, k - 1$. Denote this value as β_k. If all the possible values for x fail to satisfy the constraints for $\Phi_{s_1}, \ldots, \Phi_{s_k}$, that indicates the sequential approach fails with the specified order. Then quit the algorithm.
  Construct new objective function

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

  where $\xi_{s_1,\ldots,s_k}$ is optimal design for $(1 - \beta_k)\Phi_{s_1,\ldots,s_{k-1}}(\xi) + \beta_k\Phi_{s_k}(\xi)$. Set $k = k + 1$ and repeat Step 2, until $k = n + 1$.
- Step 3 Transfer $\Phi_{\{s_1,\ldots,s_{n+1}\}}(\xi)$ back to $\sum_{i=0}^{n} \lambda_i\Phi_i(\xi)$ according to format of Eq. (2.4) using scalar change. Then $\sum_{i=0}^{n} \lambda_i\Phi_i(\xi)$ will be the weighted optimal design problem found for the original constrained optimal design problem (2.1) with the sequential approach based on the specified order.

Table 8
Example IV: the relative efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, and ξ^*.

Design type	Efficiency			
	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

A.3. Examples

Example IV. Under the same set up as that of Example II, 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 function is

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

This indicates that only two out of the three constraints 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 8. The computational time is around 56 s.

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

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

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 ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and the constrained optimal design ξ^* under different optimal criteria are shown in Table 9. It takes 2 min and 27 s for the new algorithm to find ξ^*.

A.4. Theory and proof

Constrained optimization approach 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

Table 9
Example V: the relative efficiencies of $\xi_0^*, \xi_1^*, \xi_2^*, \xi_3^*, \xi_4^*$, and ξ^*.

Design type	Efficiency				
	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

can be written as

$$\underset{\xi \in \Xi}{\text{Maximize}}\ \Phi_0(\xi) \text{ subject to } \Phi_i(\xi) \geq c_i,\ \ i = 1, \ldots, n, \tag{A.2}$$

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.

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

Proof of Theorem 3.1. 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 (3.3) in the theorem separately.

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

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^1}), \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^1}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^0}). \end{aligned} \tag{A.3}$$

Notice that

$$\begin{aligned} &(\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) = (\mathbf{U}_S^0)^T \hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u_a^0 \hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^T \hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}), \\ &(\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^1}) = (\mathbf{U}_S^0)^T \hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u_a^0 \hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^T \hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}), \\ &(\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) = (\mathbf{U}_S^1)^T \hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u_a^1 \hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^T \hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}), \text{ and} \\ &(\mathbf{U}^1)^T \hat{\Phi}(\xi_{\mathbf{U}^1}) = (\mathbf{U}_S^1)^T \hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u_a^1 \hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^T \hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}). \end{aligned} \tag{A.4}$$

Adding up the two inequalities in (A.3) and utilizing (A.4), we have

$$(u_a^0 - u_a^1)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) + (\mathbf{U}_S^0 - \mathbf{U}_S^1)^T (\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) \geq 0. \tag{A.5}$$

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

$$(\mathbf{U}_S^0 - \mathbf{U}_S^1)^T (\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) = \sum_{i \in S} (u_i^0 - u_i^1)(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \leq 0, \tag{A.6}$$

which indicates

$$(u_a^0 - u_a^1)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) \geq 0. \tag{A.7}$$

Thus the conclusion follows. □

Proof of Theorem 3.2. Define $S_{11} = \{i | \hat{\Phi}_i(\xi_{\mathbf{U}^0}) > 0, i \in S_1\}$ and $S_{12} = \{i | \hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0, i \in S_1\}$. 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 nonnegative value set $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$ with $u_i \hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) \geq 0$ for $i \in S$. Then we have

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^+}) \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^+)^T \hat{\Phi}(\xi_{\mathbf{U}^+}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^+)^T \hat{\Phi}(\xi_{\mathbf{U}^0}). \end{aligned} \tag{A.8}$$

Then the summation of two inequalities in (A.8) returns

$$\begin{aligned}&(\mathbf{U}^0_{S_{12}}-\mathbf{U}^+_{S_{12}})^T(\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^+}))\\&+(\mathbf{U}^0_{S_{11}}-\mathbf{U}^+_{S_{11}})^T(\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^+}))\\&+(\mathbf{U}^0_{S_2}-\mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^+}))\geq 0.\end{aligned}\tag{A.9}$$

By our assumption, for any $i\in S_{12}$, we have $\hat{\Phi}_i(\xi_{\mathbf{U}^0})=0$, since $\hat{\Phi}_i(\xi_{\mathbf{U}^+})\geq 0$, thus $\hat{\Phi}_i(\xi_{\mathbf{U}^0})-\hat{\Phi}_i(\xi_{\mathbf{U}^+})\leq 0$. For u_i^+, we have 2 cases:

Case 1 $u_i^+>0$, then by assumption, $\hat{\Phi}_i(\xi_{\mathbf{U}^+})=0$, thus $(u_i^0-u_i^+)(\hat{\Phi}_i(\xi_{\mathbf{U}^0})-\hat{\Phi}_i(\xi_{\mathbf{U}^+}))=0$.
Case 2 $u_i^+=0$, then $(u_i^0-u_i^+)\geq 0$, thus $(u_i^0-u_i^+)(\hat{\Phi}_i(\xi_{\mathbf{U}^0})-\hat{\Phi}_i(\xi_{\mathbf{U}^+}))\leq 0$.

Considering these two cases, one can easily find that

$$(\mathbf{U}^0_{S_{12}}-\mathbf{U}^+_{S_{12}})^T(\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^+}))\leq 0.\tag{A.10}$$

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

Now consider set S_2, for any $i\in S_2$, we have $\hat{\Phi}_i(\xi_{\mathbf{U}^0})<0$ and $u_i^0=N_i$. Since $u_i^+\in[0,N_i)$, $(u_i^0-u_i^+)>0$. As $\hat{\Phi}_i(\xi_{\mathbf{U}^+})\geq 0$, $\hat{\Phi}_i(\xi_{\mathbf{U}^0})-\hat{\Phi}_i(\xi_{\mathbf{U}^+})<0$. One can easily see that as long as $S_2\neq\emptyset$, which also means we can find at least one $i\in S$ such that $\hat{\Phi}_i(\xi_{\mathbf{U}^0})<0$, we have

$$(\mathbf{U}^0_{S_2}-\mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^+}))<0.\tag{A.12}$$

By inequality (A.10) (A.11) (A.12), we have

$$\begin{aligned}&(\mathbf{U}^0_{S_{12}}-\mathbf{U}^+_{S_{12}})^T(\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_{12}}(\xi_{\mathbf{U}^+}))\\&+(\mathbf{U}^0_{S_{11}}-\mathbf{U}^+_{S_{11}})^T(\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^+}))\\&+(\mathbf{U}^0_{S_2}-\mathbf{U}^+_{S_2})^T(\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})-\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^+}))<0\end{aligned}\tag{A.13}$$

as long as there exists at least one $i\in S$ such that $\hat{\Phi}_i(\xi_{\mathbf{U}^0})<0$. This conflicts inequality (A.9), thus there does not exist a nonnegative value set $\mathbf{U}^+=\{\mathbf{U}_S^+,0\}$ with $u_i\hat{\Phi}_i(\xi_{\mathbf{U}^+})=0$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^+})\geq 0$ for $i\in S$ □

Proof of Theorem 4.1. Since there exists an optimal solution for the constrained optimal design problem (2.1) in manuscript, 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 are no active constraints, to most complex case, i.e., all constraints are active. If the algorithm stops early, by logics of the new algorithm procedure, the proposed design ξ^* from implementing the new algorithm will satisfy Property (2.3). Thus by Theory 2.1, the optimal design from the algorithm will be optimal to constrained problem (2.1). The only case the new algorithm may fail is: there exists some $i\in\{s|1,2,\ldots,n\}$ such that $\hat{\Phi}_i(\xi_{\mathbf{U}})<0$ when $\mathbf{U}$ is the corresponding weight found by the new algorithm after the search procedure goes through the most complex case: all constraints are assumed to be active.

Now to prove the converge of the new algorithm, one just needs to show that a feasible solution does not exist for problem (2.1) under this case.

Now suppose if there still exists a feasible solution ξ^* for problem (2.1) under this case. By Theorem 2.1, ξ^* will also be optimal to the corresponding compound design problem with weight non-negative weight vector U^*. The U^* and ξ^* will also satisfy $U^{*T}\hat{\Phi}(\xi_{\mathbf{U}})=0$ and $\hat{\Phi}(\xi_{\mathbf{U}})\geq 0$. As we mentioned above, if the algorithm fails, the algorithm has already gone through the most complex case. And for the final design $\xi_{\mathbf{U}}$ then at least one criterion $\hat{\Phi}_i(\xi_{\mathbf{U}})<0$. $\xi_{\mathbf{U}}$ and $\mathbf{U}$ will meets all the conditions stated in Theorem 3.2. Thus by Theorem 3.2, there does not exist a non-negative value set $\mathbf{U}^+$, such that $u_i\hat{\Phi}_i(\xi_{\mathbf{U}^+})=0$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^+})\geq 0$. This conflicts the assumption that there exists the corresponding weight U^*. Thus this also conflicts the assumption that the constrained problem (2.1) can still have feasible solution when the new algorithm fails. Thus a feasible solution does not exist for problem (2.1) if the new algorithm fails. □

References

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

Cao, Y., Smucker, B.J., Robinson, T.J., 2015. On using the hypervolume indicator to compare Pareto fronts: Applications to multi-criteria optimal experimental design. JSPI 160, 60–74.

Cheng, Q., Majumdar, D., Yang, M., 2016. On multiple-objective nonlinear optimal designs. In: Kunert, J., Mller, C., Atkinson, A. (Eds.), mODA 11 - Advances in Model-Oriented Design and Analysis. In: Contributions to Statistics, pp. 63–70.

Clyde, M., Chaloner, K., 1996. The equivalence of constrained and weighted designs in multiple objective design problems. J. Amer. Statist. Assoc. 91, 1236–1244.

Cook, D., Fedorov, V., 1995. Constrained optimization of experimental design. Statistics 26, 129–178.

Cook, R.D., Wong, W.K., 1994. On the equivalence of constrained and compound optimal designs. J. Amer. Statist. Assoc. 89, 687–692.

Harmon, R., Benkova, E., 2017. Barycentric algorithm for computing D-optimal size- and cost-constrained designs of experiments, 80:201225. Metrika 80, 201–225.

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

Kao, M.-H., Mandal, A., Stufken, J., 2012. Constrained multi-objective designs for functional mri experiments via a modified nondominated sorting genetic algorithm. J. Roy. Statist. Soc. Ser. C 61, 515–534.

Mandal, S., Torsney, B., Carriere, K.C., 2005. Constructing optimal designs with constraints. JSPI 128, 609–621.

Mandal, A., Wong, W.K., Yu, Y., 2015. Algorithmic searches for optimal designs. In: Bingham, D., Dean, A., Morris, M., Stufken, J. (Eds.), Handbooks on Modern Statistical Methods, Design of Experiments. pp. 755–783.

Mikulecká, J., 1983. On a hybrid experimental design. Kybernetika 19 (1), 1–14.

Notari, R.E., 1980. Biopharmaceutics and Clinical Pharmacokinetics. Marcel Dekker, New York and Basel.

Rosenberger, W.F., Grill, S.E., 1997. A sequential design for psychophysical experiments: an application to estimating timing of sensory events. Stat. Med. 16, 2245–2260.

Sagnol, G., Harman, R., 2015. Computing exact d-optimal designs by mixed integer second order cone programming. Ann. Statist. 43(5), 2198–2224.

Vandenberghe, L., Boyd, S., Wu, S., 1998. Determinant maximization with linear matrix inequality constraints. SIAM. J. Matrix Anal. 19, 499–533.

Yang, M., Biedermann, S., Tang, E., 2013. On optimal designs for nonlinear models: a general and efficient algorithm. J. Amer. Statist. Assoc. 108, 1411–1420.

Yu, Y., 2011. D-optimal designs via a cocktail algorithm. Stat. Comput. 21, 475–481.