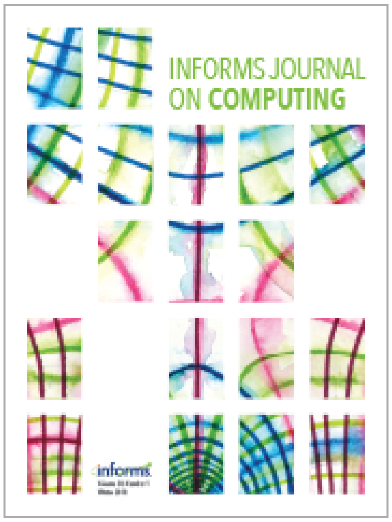


INFORMS Journal on Computing

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Domination Measure: A New Metric for Solving Multiobjective Optimization

Joshua Q. Hale, Helin Zhu, Enlu Zhou

To cite this article:
Joshua Q. Hale, Helin Zhu, Enlu Zhou (2020) Domination Measure: A New Metric for Solving Multiobjective Optimization. INFORMS Journal on Computing 32(3):565-581. https://doi.org/10.1287/ijoc.2019.0920

informs
http://pubsonline.informs.org/journal/ijoc

INFORMS JOURNAL ON COMPUTING
Vol. 32, No. 3, Summer 2020, pp. 565–581
ISSN 1091-9856 (print), ISSN 1526-5528 (online)

Domination Measure: A New Metric for Solving Multiobjective Optimization

Joshua Q. Hale,[a] **Helin Zhu,**[b] **Enlu Zhou**[c]

[a] Intel Corporation, Chandler, Arizona 85226; [b] Uber Technologies, Inc., San Francisco, California 94103; [c] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332

Contact: joshua.q.hale@intel.com (JQH); zhuhelin1990@gmail.com (HZ); enlu.zhou@isye.gatech.edu, http://orcid.org/0000-0001-5399-6508 (EZ)

Received: October 4, 2016
Revised: November 29, 2017; March 19, 2019
Accepted: June 25, 2019
Published Online in Articles in Advance: January 30, 2020

https://doi.org/10.1287/ijoc.2019.0920

Abstract. For general multiobjective optimization problems, the usual goal is finding the set of solutions not dominated by any other solutions, that is, a set of solutions as good as any other solution in all objectives and strictly better in at least one objective. In this paper, we propose a novel performance metric called the *domination measure* to measure the quality of a solution, which can be intuitively interpreted as the probability that an arbitrary solution in the solution space dominates that solution with respect to a predefined probability measure. We then reformulate the original problem as a stochastic and single-objective optimization problem. We further propose a model-based approach to solve it, which leads to an ideal version algorithm and an implementable version algorithm. We show that the ideal version algorithm converges to a set representation of the global optima of the reformulated problem; we demonstrate the numerical performance of the implementable version algorithm by comparing it with numerous existing multiobjective optimization methods on popular benchmark test functions. The numerical results show that the proposed approach is effective in generating a finite and uniformly spread approximation of the Pareto optimal set of the original multiobjective problem and is competitive with the tested existing methods. The concept of domination measure opens the door for potentially many new algorithms, and our proposed algorithm is an instance that benefits from domination measure.

History: This paper has been accepted by Alice Smith, Editor-in-Chief, and Bruno Tuffin, Area Editor, Simulation.
Funding: This work was supported by the National Science Foundation [Grants CMMI-1413790 and CAREER CMMI-1453934] and the Air Force Office of Scientific Research [Grant YIP FA-9550-14-1-0059].
Supplemental Material: The online supplement is available at https://doi.org/10.1287/ijoc.2019.0920.

Keywords: multiobjective optimization • model-based optimization • domination measure • hypervolume

1. Introduction

Problems that require optimizing several objectives concurrently are known as multiobjective optimization problems. This type of problem arises in many real-world applications, including construction science (Le Riche et al. 2003), economics (Toffolo and Lazzaretto 2002), medical treatments (Qasem and Shamsuddin 2011), scheduling (Minella et al. 2008), and logistics (Lee et al. 2008), in which incommensurable and conflicting objectives need to be optimized. Therefore, it is often unlikely to have a solution that optimizes all objectives simultaneously. A more reasonable goal is to obtain a set of solutions, where the quality of each solution is incomparable without any prior knowledge of preference. These solutions are known as *Pareto optimal* solutions, which are “optimal” in the sense that no other solutions in the solution space dominate them, that is, having better objective values for all objectives. The set of Pareto optimal solutions is called the Pareto optimal set, and its image in the objective space is called the *Pareto front*. Then the goal is to find a finite (and hopefully evenly spread) representation of the Pareto optimal set.

Numerous methods have been developed to achieve such goal for general multiobjective optimization problems, among which perhaps the most popular ones are evolutionary algorithms (Deb 2001) that use iterative selection, mutation and crossover operations to generate multiple Pareto optimal solutions in parallel. Other methods include stochastic search methods (Zabinsky 2013) that choose candidate solutions at random and improve the way those candidate solutions are selected in each iteration, particle swarm methods (Mete and Zabinsky 2014) that keep a population of potential solutions (particles) which are manipulated by a velocity vector, which changes the position of the particles at each iteration, and metaheuristic procedures (Köksalan and Phelps 2007, Molina et al. 2007). Furthermore, other derivative-free methods described in Nam and Park (2000) and Custódio et al. (2011) also have been implemented in the multiobjective domain.

The majority of the aforementioned methods can be classified as instance-based methods, as they maintain a population of candidate solutions, and new candidate solutions are generated by only considering previously generated solutions. In contrast, model-based search algorithms generate candidate solutions from parameterized sampling distributions that are iteratively updated using previous candidate solutions. They are effective at optimizing functions that lack structural properties, such as differentiability and convexity. Although model-based algorithms mostly have been used to solve single-objective optimization problems, some methods incorporate the cross entropy (CE) (Rubinstein 2001) method which is a model-based approach to solve multiobjective problems (see, e.g., Unveren and Acan 2007 and Bekker and Aldrich 2011). In Unveren and Acan (2007) samples are generated from a fixed number of sampling distributions, which allows the algorithm to be very fast and simple to implement. However, because a fixed number of sampling distributions are used, this method may have difficulty in constructing a sampling distribution that has the majority of its mass on the isolated points of the Pareto optimal set. Bekker and Aldrich (2011) propose a histogram approach in which candidate solutions are drawn independently along each dimension of the solution space. Although this method performs very well on some problems, it could have difficulty in capturing the entire Pareto optimal set that consists of highly correlated solutions.

Most of the methods mentioned above are designed for optimizing multiple objectives that are deterministic. A number of methods have also been developed to solve the stochastic version of this problem (see, e.g., Lee et al. 2010 and Feldman et al. 2015). The details of these methods are beyond the scope of this paper, because we are focused on deterministic objectives.

In view of the existing methods in the literature, a method that explores the dominance relationship among solutions by simple quality metrics is still lacking. Incorporating such metrics reduces problem dimensionality because the multidimensional objective space is mapped onto a single-dimensional one through direct comparisons among solutions. As a result, the original multiobjective problem is transformed into a single-objective one, in which the objective is to find solutions that optimize a particular quality metric. As pointed out by Zitzler et al. (2003), a reduction of problem dimensionality will inevitably cause the loss of information. That is, generally the global optimal set of the reformulated problem will not be the same as the Pareto optimal set. In particular, Zitzler et al. (2003) prove that in order for a metric to retain the Pareto dominance relation, the dimension of the metric and the objective space must be equivalent. A multitude of methods employ the scalarization technique (see Santiago et al. 2014). For example, a quality metric can be derived by a weighted aggregation of the objective functions. The issue with this quality metric is that a single choice of weights leads to at most one point in the Pareto optimal set. Although multiple points could be obtained via changing the weights, the final approximation could be unsatisfactory because a uniform spread of the weighting coefficients does not necessarily produce a uniform spread of Pareto optimal solutions. Moreover, this technique is infeasible for problems with a large amount of objectives because the total number of weight combinations grows exponentially with respect to (w.r.t) the number of objectives.

In this paper, we introduce a new parameter-free and unary performance metric to measure the quality of solutions, termed as *domination measure*. On a high level, the domination measure of a solution can be viewed as the measure of the region in the solution space that dominates that solution w.r.t. a predefined probability measure.

Using the domination measure as a performance metric brings many advantages. Foremost, unlike the aforementioned scalarization methods, this metric does not require any tuning of parameters. Moreover, domination measure is a rigorous quantification for the quality of a solution; the lower the domination measure, the better the solution. Finally, if a solution is Pareto optimal, then it has a domination measure of zero because no solution dominates it. Although theoretically the set of solutions with domination of zero is not equivalent to the Pareto optimal set, it is a sufficiently good approximation in the sense that no solution dominates it almost surely w.r.t. the underlying predefined probability measure.

By optimizing on domination measure, we are able to transform the original multiobjective optimization problem into a single-objective stochastic optimization problem, in which the domination measure could be viewed as the expectation of an indicator function on the dominance relation w.r.t. an induced probability measure. Then the goal is to find the set of global optima to the reformulated problem, that is, the set of solutions with domination measure of zero. In practice, often a finite and evenly spread representation of such set is sufficient. Note that while we benefit from a significant reduction in problem complexity, we make the compromise from finding Pareto optimal solutions to finding solutions with domination measure of zero.

To solve the reformulated problem, we propose a model-based approach that generates multiple global optima. The idea is similar to the one used in the CE method (Rubinstein 2001), in which a sequence of parameterized sampling distributions and a sequence of reference distributions are simultaneously tracked, and the sampling parameter is iteratively updated by minimizing the Kullback-Leibler (KL) divergence between the pair of distributions.

The novelty in our approach lies in that we use a mixture sampling distribution with an adaptive number of components from the same parameterized family of densities, so that the mixture sampling distribution captures the Pareto optimal set and eventually converges to multiple Pareto optimal solutions.

The proposed idea leads to two versions of the algorithm: an ideal version that ignores the Monte Carlo sampling error, and an implementable version that uses Monte Carlo sampling and clustering in the algorithm. For the ideal version, we show that when the mixture sampling distributions belong to the so-called "exponential family," for any solution in the Pareto optimal set, there exists a sequence of sampling distributions that converges to it. In the implementable version, we sample from the mixtures sampling distribution. Moreover, to ensure the samples lead to evenly spread solutions that cover the Pareto optimal set, we further introduce a clustering method to cluster the samples before updating the mixture sampling distribution. We show empirically that the implementable version algorithm is competitive with many existing multiobjective optimization methods by testing them on several benchmark problems. In particular, we observe that in all tested cases, our approach is able to generate solutions that are almost evenly spread on the Pareto optimal set by a user-specified threshold distance. In summary, the contributions in this paper are as follows.

- For multiobjective optimization problems, we introduce a novel performance metric called domination measure to determine the quality of a solution, based on which we reformulate the original problem into a stochastic and single-objective one that aims to minimize the domination measure. This reformulation reduces the complexity of the problem, and allows potential adaptation of algorithms for single-objective stochastic optimization to solve multiobjective optimization.
- We propose a novel model-based approach to solve the reformulated problem, leading to an ideal version algorithm that possesses nice convergence properties and an implementable version algorithm that performs well numerically.
- We demonstrate in numerical experiments that our proposed approach produces a finite and evenly spread representation of the Pareto optimal set and performs competitively to or outperforms many existing approaches.

A performance metric termed "hypervolume" and proposed by Zitzler and Thiele (1998) is analogous to the domination measure in the objective space. The hypervolume of a set of points is the total size of the objective space that is dominated by that set w.r.t. a reference point. Hypervolume is desirable for assessing the performance of a solution set because it has the ability to measure how close solutions are to the Pareto front and how evenly spread the solutions are in the objective space. The hypervolume is also used in many evolutionary algorithms as a performance metric (Fleischer 2003, Beume et al. 2007, Bader and Zitzler 2011). Although the domination measure may seem to be the decision-space version of the hypervolume, both methods have distinct characteristics. In particular, the hypervolume is greatly dependent on the choice of the reference point (Pal and Bandyopadhyay 2016). The impact of the reference point on the hypervolume has not been fully understood, and in practice the hypervolume literature does not agree on how to choose the reference point (Auger et al. 2012). Unlike hypervolume, the domination measure calculation does not depend on any fixed parameter but instead only depends on objective evaluations. Similar to the domination measure, exact calculation of the hypervolume is extremely computationally expensive. Although both metrics can be approximated via Monte Carlo sampling, hypervolume approximation is less straightforward. Specifically, the domination measure of a solution is approximated by sampling in the solution space and calculating the "proportion" of the points that dominates that solution. In contrast, hypervolume approximation requires constructing a hyperrectangle as the sampling area in the objective space (Bader et al. 2010). Constructing the hyperrectangle can be a challenging problem because, in most cases, doing so requires determining the optimal value for each objective separately. Lastly, the hypervolume could prefer convex regions of the objective space to concave regions (Zitzler et al. 2007). Whereas the domination measure is not sensitive to any area of the objective or solution space. To the best of our knowledge, our work is the first to propose the domination measure as a performance metric for solving multiobjective problems.

The remainder of this paper is organized as follows. In Section 2, we establish the concept and common properties of domination measure under the umbrella of multiobjective optimization and reduce the original problem into a stochastic and a single-objective one. In Section 3, we present a model-based approach to solve the reformulated problem and describe the ideal version and the implementable version algorithms. In Section 4, we conduct numerical experiments to demonstrate the effectiveness and advantages of the proposed approach by comparing it with existing approaches. Finally, conclusions and future directions of research are given in Section 5.

2. Multiobjective Optimization and Domination Measure

A general multiobjective problem consists of minimizing (or maximizing) multiple objectives over a defined solution space, which can be formulated as follows:

$$\min_{x \in \mathscr{X}} \quad \mathbf{f}(x) = \left\{ f_1(x), f_2(x), \ldots, f_n(x) \right\} \tag{1}$$

where $\mathscr{X}$ denotes the solution space, and $\{f_i(\cdot):\mathscr{X}\to\mathbb{R}, i=1,\ldots,n\}$ are scalar functions. Without loss of generality, we assume $\mathscr{X}$ is a *bounded* subset of $\mathbb{R}^d$. Here min in (1) means that we are searching for a solution $x\in\mathscr{X}$ that minimizes all the $f_i(\cdot)$.

It is rarely the case that there exists a solution in $\mathscr{X}$ that minimizes all the objectives simultaneously. Thus, a compromised goal is to find all the solutions that are not dominated by any other solutions in $\mathscr{X}$ in terms of the objective values. Specifically, we say a solution $x \in \mathscr{X}$ is *(Pareto) dominated* by another solution $y \in \mathscr{X}$ if $f_i(y) \le f_i(x)$ for all $i = 1, \ldots, n$, and there exists at least one $j \in \{1, \ldots, n\}$ such that $f_j(y) < f_j(x)$. Note that Pareto dominance is a (strict) partial order defined on $\mathscr{X}$, because it is irreflexive, asymmetric, and transitive. For simplicity, we use $y \prec_d x$ to denote that y dominates x. Then we say a solution $x \in \mathscr{X}$ is *Pareto optimal* if no other solution in $\mathscr{X}$ dominates x. That is, $x \in \mathscr{X}$ is Pareto optimal if and only if $y \nprec_d x, \forall y \in \mathscr{X}$.

Our goal is to find a finite (and hopefully evenly spread) representation of the *Pareto optimal set*—$\mathscr{P}^*$. Without loss of generality, we assume there exists at least one Pareto optimal solution, that is, $\mathscr{P}^* \neq \emptyset$.

In general, finding the Pareto optimal set or its representation is difficult because the dominance relationship $\prec_d$ is a partial order defined on the solution space. Essentially one needs to solve a combinatorial problem over the (often continuous) solution space. Thus, many existing approaches in the literature try to solve various approximations or reductions of the original multiobjective problem. In particular, one class of approaches is to reformulate the original problem as a single-objective one (e.g., through a weighting scheme on the objective functions) and apply suitable algorithms. However, it usually suffers from drawbacks, such as (1) certain information of the original problem (e.g., the dominance relationship) might be lost and (2) the weighting scheme might be difficult to choose.

These observations motivate us to develop a multiobjective to single-objective transformation that is more self-contained and interpretable. Specifically, we propose a performance metric on $\mathscr{X}$ called domination measure that quantifies the dominance relationship between an arbitrary solution in $\mathscr{X}$ and all other solutions.

2.1. Domination Measure

As mentioned previously, domination measure is constructed on the support space $\mathscr{X}$ of problem (1), and it characterizes the Pareto dominance relationship between an arbitrary solution in $\mathscr{X}$ and all other solutions. In particular, the construction is on top of an arbitrary nonzero Radon measure $\nu(\cdot)$ that is predefined on $\mathscr{X}$, as described in the following Definition 1.

Definition 1 (Domination Measure)**.** Assume $\mathscr{X} \subset \mathbb{R}^d$ in problem (1) is bounded and Lebesgue measurable with nonzero measure, and on $\mathscr{X}$ there exists a nonzero Radon measure $\nu(\cdot)$. Further assume $\forall\ x \in \mathscr{X}$ the set of solutions that dominates x, denoted by $\mathscr{D}_x$, is ν-measurable. Then, $\forall\ x \in \mathscr{X}$ the domination measure $D(x)$ of x is defined as

$$D(x) \triangleq \frac{\nu(\mathscr{D}_x)}{\nu(\mathscr{X})}, \tag{2}$$

where, for an arbitrary ν-measurable set $\mathscr{A}$, $\nu(\mathscr{A})$ denotes the measure of $\mathscr{A}$ w.r.t. $\nu(\cdot)$.

Note that in (2) the denominator $\nu(\mathscr{X})$ is finite because $\mathscr{X}$ is bounded. It is also nonzero because by definition $\nu(\cdot)$ is a nonzero measure. Therefore, $\forall\ x \in \mathscr{X}, D(x)$ is well defined. We also make the following observations.

(a) One obvious choice of $v(\cdot)$ is the underlying Lebesgue measure. Then the resultant domination measure $D(x)$ could be roughly interpreted as the "proportion" of the solutions within $\mathscr{X}$ that dominates x in "volume."

(b) Defining domination measure under an arbitrary "well-behaved" Radon measure enables us to put different weights on the solutions prior to the optimization. For example, if one has prior knowledge that certain solutions are undesirable, then naturally she/he would want to eliminate solutions that are dominated by these solutions. By constructing a Radon measure with higher weights on these solutions, any solution dominated by them would have a larger domination measure. Thus, the preferences could be taken into account quantitatively.

(c) By assuming $\mathscr{X}$ has a nonzero Lebesgue measure, we implicitly assume that the solution space is at least uncountable, or even continuous. When $\mathscr{X}$ is discrete and finite, one could construct the underlying measure $\nu(\cdot)$ by imposing a (uniform) probability distribution on $\mathscr{X}$.

To the best of our knowledge, this work is the first to establish the concept of domination measure among the literature of multiobjective optimization. Some of the straightforward extensions and properties about domination measure could be immediately established as well.

- $\forall\ x \in \mathscr{X}$, we have $0 \le D(x) \le 1$. This is because $\mathscr{D}_x \subseteq \mathscr{X}$, and, thus, $0 \le \nu(\mathscr{D}_x) \le \nu(\mathscr{X})$.
- If $y \prec_d x$, then we have $D(y) \le D(x)$. This is because any solution that dominates y will also dominate x, and, thus, $\mathscr{D}_y \subseteq \mathscr{D}_x$ and $\nu(\mathscr{D}_y) \le \nu(\mathscr{D}_x)$.

Intuitively, the smaller $D(x)$ is, the less likely x is dominated by another solution in $\mathscr{X}$. In particular, we could easily show that the domination measure of an arbitrary Pareto optimal solution $x^* \in \mathscr{P}^*$ is zero.

Lemma 1. *For any Pareto optimal solution $x^* \in \mathscr{P}^*$, its domination measure $D(x^*) = 0$.*

The proof of Lemma 1 is straightforward, noting that no point in $\mathscr{X}$ dominates x^*, and thus $\mathscr{D}_{x^*} = \emptyset$ and $\nu(\mathscr{D}_{x^*}) = 0$.

However, the other direction of the statement in Lemma 1 might not be true. That is, a solution with domination measure of zero might not be Pareto optimal, as illustrated in the following example.

Example 1. Consider a simple problem with two objectives on a two-dimensional solution space $[0,1] \times [0,1] \subset \mathbb{R}^2$, as displayed in Figure 1. The two objectives are $f_1(x_1, x_2) = x_1$ and $f_2(x_1, x_2) = x_2$. Therefore, (f_1, f_2) is a mapping from $[0,1] \times [0,1]$ to itself. Note that although $(0,0)$ is the only Pareto optimal solution, all solutions in the set $\{(x_1, x_2) : x_1 = 0, \text{ or } x_2 = 0\}$ (highlighted in blue color) have domination measure of zero. For instance, the point $(0.4, 0)$ is dominated by all points in the set $\mathscr{B} := \{(x_1, x_2) : x_1 \in [0, 0.4), x_2 = 0\}$, but the Lebesgue measure of $\mathscr{B}$ is zero.

Lemma 1 and Example 1 provide us with an alternative goal: finding the set of all points with domination measure of zero or its finite representation, which is softer than finding the Pareto optimal set.

To this end, let us use $\mathscr{D}^*$ to denote such a set, that is, $\mathscr{D}^* := \{x \in \mathscr{X} : D(x) = 0\}$. Then our goal in the remainder of the paper is to *find a finite (and hopefully evenly spread) representation of* $\mathscr{D}^*$. Note that a direct consequence of Lemma 1 is that $\mathscr{P}^* \subseteq \mathscr{D}^*$. Given the assumption that $\mathscr{P}^* \neq \emptyset$, it follows that $\mathscr{D}^* \neq \emptyset$.

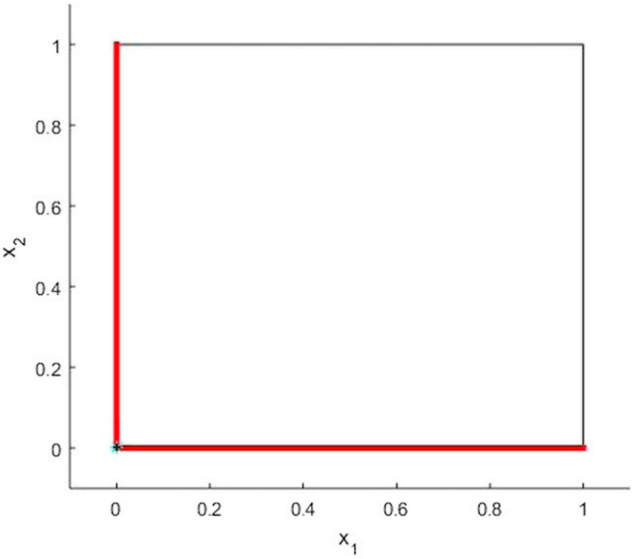
There are also cases in which $\mathscr{P}^* = \mathscr{D}^*$. For example, one trivial case is that when the solution set $\mathscr{X}$ is finite, as demonstrated in the following example.

Example 2 (Mete and Zabinsky 2014)**.** Consider a biobjective optimization problem in which

$$\begin{cases} f_1(x) = 0.001x(x-10)(x-60)(x-100) + 1000, \\ f_2(x) = 0.001x(x-70)(x-100)(x-200) + 6000. \end{cases}$$

The solution space is $\mathbb{Z} \cap [0, 100]$, that is, the set of all the integers between 0 and 100, which is a finite set.

Figure 1. (Color online) Illustration of Domination Measure



Without going into details, the Pareto optimal set is $\mathscr{P}^* = \mathbb{Z} \cap ([5, 25] \cup [60, 85])$, and it is highlighted in Figure 2.

Directly locating the Pareto optimal set using function values (see Figure 2(a)) is difficult, whereas the Pareto optimal set is easily identifiable using domination measure (see Figure 2(b)). That is, the solutions with domination measure of zero are exactly the Pareto optimal set for this problem.

Remark 1. We could extend the definition of the domination measure from a point to a finite set. Specifically, assume $\mathscr{A} \subset \mathscr{X}$ is a finite set, then we could define its domination measure $D(\mathscr{A})$ as the ν-measure of the set consisting of all the points that dominate every point in $\mathscr{A}$. Denoting such a set by $\mathscr{D}_{\mathscr{A}}$, and we immediately have $\mathscr{D}_{\mathscr{A}} = \cap_{y \in \mathscr{A}} \mathscr{D}_y$, and

$$D(\mathscr{A}) = \frac{\nu(\cap_{y \in \mathscr{A}} \mathscr{D}_y)}{\nu(\mathscr{X})} \leq \frac{\nu(\mathscr{D}_x)}{\nu(\mathscr{X})} = D(x), \forall\ x \in \mathscr{A}.$$

Similarly, for any two finite sets $\mathscr{A}, \mathscr{B} \subseteq \mathscr{X}$, we have

$$D(\mathscr{A} \cup \mathscr{B}) = \nu(\mathscr{D}_{\mathscr{A}} \cap \mathscr{D}_{\mathscr{B}})/\nu(\mathscr{X}) \text{ and}$$
$$D(\mathscr{A} \cap \mathscr{B}) = \nu(\mathscr{D}_{\mathscr{A}} \cup \mathscr{D}_{\mathscr{B}})/\nu(\mathscr{X}).$$

2.2. Reformulation as a Single-Objective Problem

Note that we could reformulate (2) as

$$\begin{aligned} D(x) &= \frac{\nu(\mathscr{D}_x)}{\nu(\mathscr{X})} = \frac{\int_{\mathscr{X}} \mathbb{1}\{y \prec_d x\} \nu(dy)}{\int_{\mathscr{X}} \nu(dy)} \\ &= \int_{\mathscr{X}} \mathbb{1}\{y \prec_d x\} \mathscr{U}(dy) \stackrel{\Delta}{=} \mathbb{E}_{\mathscr{U}(y)}[\mathbb{1}\{y \prec_d x\}], \end{aligned} \qquad (3)$$

where $\mathbb{1}\{E\} = 1$ if the event E is true and $\mathbb{1}\{E\} = 0$ otherwise, $\mathscr{U}(\cdot)$ is the probability measure on $\mathscr{X}$ induced by $\nu(\cdot)$, and $\mathbb{E}_{\mathscr{U}}[\cdot]$ denotes the expectation w.r.t. $\mathscr{U}(\cdot)$. To ease the derivation, hereafter we assume $\nu(\cdot)$ is the underlying Lebesgue measure, and, thus, $\mathscr{U}(\cdot)$ is the uniform probability measure on $\mathscr{X}$. We also point out the derivation could be easily generalized to the case of non-Lebesgue measure.

Therefore, we aim to solve the single-objective stochastic optimization problem:

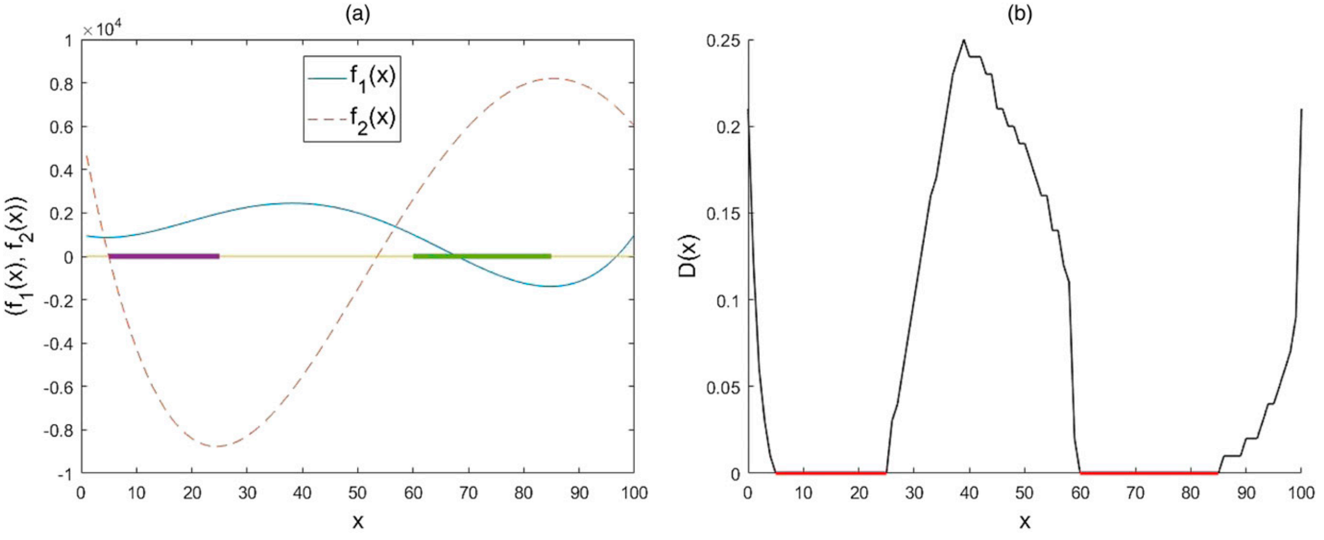
$$\mathscr{D}^* = \arg\min_{x \in \mathscr{X}} D(x) = \mathbb{E}_{\mathscr{U}(y)}[\mathbb{1}\{y \prec_d x\}],$$

or, equivalently,

$$\mathscr{D}^* = \arg\max_{x \in \mathscr{X}} -D(x) = \mathbb{E}_{\mathscr{U}(y)}[-\mathbb{1}\{y \prec_d x\}]. \qquad (4)$$

To compute $D(x)$ exactly for an arbitrary $x \in \mathscr{X}$, we need to compare its dominance relationship with all solutions almost surely (a.s.) in $\mathscr{X}$. This is often expensive or infeasible, especially when $\mathscr{X}$ is not finite. An alternative approach is to apply Monte Carlo sampling by drawing finite independent and identically

Figure 2. (Color online) Pareto Optimal Set by Function Values and Domination Measure



Pareto Optimal Set by Function Values

Pareto Optimal Set by Domination Measure

distributed (i.i.d.) sample solutions w.r.t. the underlying probably measure $\mathscr{U}(\cdot)$, and estimate the expectation in (4) by the sample average. This approach possesses at least two advantages: first, Monte Carlo simulation is straightforward to implement and scales well with the dimension of underlying solution space; second, *for the case of estimating domination measure, it is even more efficient because all the samples will be used to compare against each other, which yields an estimate of domination measure for every sample.* The general procedure is as follows.

1. Draw samples $\{x^1, \dots, x^N\} \overset{i.i.d.}{\sim} \mathscr{U}(\cdot)$ from $\mathscr{X}$.
2. Calculate

$$\widetilde{D}(x^i) \triangleq \frac{1}{N}\sum_{j=1}^{N} \mathbb{1}\{x^j \prec_d x^i\} \tag{5}$$

as the estimate of $D(x^i)$.

2.3. Optimizing the Domination Measure

Problem (4) involves a stochastic objective function, so one could apply standard optimization algorithms, such as stochastic gradient descent. However, note that these algorithms typically converge to local optima and have certain requirements on the structure of the problem, such as differentiability and convexity of the objective function; furthermore, they are suited for tracking the trajectory of a single solution, which is not sufficient because we are seeking a set of global optima. Although multiple candidate optimal solutions could be generated by a multistart implementation, it is generally difficult to design the implementation in a way such that those solutions form a good (evenly spread) representation of the global optima.

Alternatively, one could use gradient-free optimization techniques, such as model-based methods, that draw from and update the sampling distribution iteratively based on function evaluations, for example, the CE method, the model reference adaptive search (MRAS) (Hu et al. 2007), and the gradient-based adaptive stochastic search (GASS) (Zhou and Hu 2014). A typical model-based method has the following advantages for optimizing domination measure:

- It imposes minimum requirements on the problem structure.
- It inherits the advantage of Monte Carlo simulation, as it scales with the dimension of the solution space and efficiently uses samples in estimating the domination measure.
- It is able to generate good representations of the optimal optima set by appropriately selecting the family of sampling distributions, such as a mixture Gaussian family of densities.

Motivated by these advantages, next we propose a specific model-based algorithm by modifying the popular CE method (see Rubinstein 2001).

3. A Model-Based Approach

The main idea of a typical model-based method is to introduce a sampling distribution, which often belongs to a parameterized family of densities, over the solution space, and iteratively update the parameters of the sampling distribution by generating and evaluating candidate solutions. Specifically, the methods iteratively carry out the following two steps:

1. Generate candidate solutions according to the sampling distribution.
2. Based on the evaluation of the candidate solutions, update the parameter of the sampling distribution.

Note that in step 2 the updating rule controls the balance between exploration and exploitation of the solution space. The hope is to have the sampling distribution more and more concentrated on the promising region of the solution space where the optimal solutions are located, and eventually become a degenerate distribution on one of the global optima.

In principle, we could directly apply CE for the simulation optimization problem (4) if the goal is to find one global optimal solution which corresponds to a degenerate distribution. To accommodate the need for finding a good representation of $\mathscr{D}^*$ that consists of multiple global optima, we adopt the general idea of tracking both reference distributions and sampling distributions in CE-like methods and develop new updating schemes for those distributions. The hope is that (1) multiple global optima in $\mathscr{D}^*$ could be generated; (2) they form a good representation of $\mathscr{D}^*$ that are evenly spread; and (3) certain convergence guarantee could be derived.

First, instead of using a single distribution from a parameterized family as the sampling distribution, we propose to use a mixture one with equal weight on each component from the same family. The hope is that each component will concentrate on a promising region of the solution space and explore that region exclusively, and eventually become a degenerate distribution on one global optimal solution. Second, the number of components is determined adaptively so that all the promising regions are explored. Third, the updating scheme on the parameters of the mixture sampling distribution needs to be designed in a way such that the resulting degenerate distributions are evenly spread.

We also adopt the commonly used parameterized family of densities, which is the exponential family defined as follows.

Definition 2 (Exponential Family of Densities)**.** A parameterized family $\{g(x;\theta) : \theta \in \Theta\}$ is an exponential family of densities if it satisfies

$$g(x;\theta) = \exp\{\theta^T\Gamma(x) - \eta(\theta)\}, \tag{6}$$

where θ represents the parameter, $\Gamma(x) = [\Gamma_1(x),\ldots,\Gamma_{d_\theta}(x)]^T$ is the vector of sufficient statistic, $\eta(\theta) = \ln\{\int \exp(\theta^T\Gamma(x))dx\}$ is the normalization factor, and $\Theta = \{\theta : |\eta(\theta)| < \infty\}$ is the natural parameter space with a nonempty interior. We assume that $\Gamma(\cdot)$ is a continuous mapping.

For example, a common probability distribution that belongs to the exponential family is the Gaussian distribution, in which θ consists of parameters that depend on the mean vector and the covariance matrix. The advantage of using the exponential family is that the intermediate KL-divergence minimization problem could be solved analytically.

Specifically, suppose $h(x)$ is a reference sampling distribution of interest, then the corresponding sampling distribution $g(x;\theta_*)$ could be found via

$$\begin{aligned}\theta_* \triangleq \arg\min_{\theta\in\Theta} KL(h(\cdot), g(\cdot;\theta)) &= \arg\min_{\theta\in\Theta} \mathbb{E}_h\left[\ln\frac{h(X)}{g(X;\theta)}\right]\\ &= \arg\max_{\theta\in\Theta} \mathbb{E}_h[\ln g(X;\theta)],\end{aligned} \tag{7}$$

where $\mathbb{E}_h[\cdot]$ denotes the expectation w.r.t. $h(\cdot)$ and X represents the corresponding random variable. Thanks to the nice properties of exponential families, (7) admits a unique optimal solution θ_* that satisfies the first-order condition

$$\int\left(\Gamma_j(x) - \frac{\int \Gamma_j(x)\exp(\theta_*^T\Gamma(x))dx}{\int \exp(\theta_*^T\Gamma(x))dx}\right)h(x)dx = 0, \quad j = 1,\ldots,d_\theta,$$

or, equivalently,

$$\mathbb{E}_h[\Gamma(X)] = \mathbb{E}_{\theta_*}[\Gamma(X)], \tag{8}$$

where $\mathbb{E}_\theta[\cdot]$ denotes the expectation w.r.t. $g(x;\theta)$.

In practice, finding the exact value of θ_* can be difficult because $\mathbb{E}_h[\Gamma(X)]$ usually does not admit a closed-form expression; however, a good estimate of θ_* could be obtained via the principle of importance sampling, noting that

$$\mathbb{E}_{\theta_*}[\Gamma(X)] = \mathbb{E}_h[\Gamma(X)] = \mathbb{E}_{\theta'}\left[\frac{\Gamma(X)h(X)}{g(X;\theta')}\right],$$

and i.i.d. samples could be drawn according to $g(\cdot;\theta')$ to estimate the expectation on the right-hand side.

3.1. An Ideal Version Algorithm

We first present an ideal version algorithm of the proposed approach. Although it is not directly implementable in practice, its merit lies in unveiling the mathematical intuition of an implementable algorithm to be introduced later.

Recall that in the modified CE algorithm, we need to (1) determine the number of components in the mixture sampling distribution and (2) determine/update the parameters for each component. In the ideal version algorithm, we use a heuristic to determine the number of components that yield a convergence guarantee and propose simple reference distributions that converge to degenerate distributions.

Let us use $\mathbf{g}_k(x)$, where

$$\mathbf{g}_k(x) = \frac{1}{I_k}\sum_{i=1}^{I_k} g(x;\theta_{i,k}), \tag{9}$$

to denote the equally weighted mixture sampling distribution at iteration k, where I_k is the number of components and $\theta_{i,k}$ is the parameter of the ith component. The corresponding population of reference distributions is denoted by $\{h_{i,k}(x) : i = 1, \ldots, I_k\}$. We also introduce a decreasing sequence of reference values, denoted by $\{\gamma_k : k = 1, \ldots\}$, that satisfies

$$\gamma_k \in (0,1], \;\; \gamma_{k+1} < \gamma_k, \;\; \text{and} \lim_{k\to\infty} \gamma_k = 0. \qquad (10)$$

Furthermore, let

$$D_k := \left\{x \in \mathscr{X} : D(x) \le \gamma_k\right\} \qquad (11)$$

be the set of solutions with domination measure values below γ_k. Then we have $D_{k+1} \subseteq D_k$, which means the sequence of sets $\{D_k : k = 1, \ldots\}$ is nonincreasing. Therefore, its limit exists. In particular, $\lim_{k\to\infty} D_k = \cap_{k=1}^{\infty} D_k = \mathscr{D}^*$, where recall that $\mathscr{D}^*$ is the set of optimal solutions to the reformulated problem (4).

The number of components I_k is determined as follows. Assume D_k is $\mathscr{U}$-measurable, then we introduce a partition on the set D_k with I_k elements, denoted by $\pi_k := \{D_{1,k}, \ldots, D_{I_k,k}\}$, such that each partitioning set $D_{i,k}$ is $\mathscr{U}$-measurable,

$$D_k = \cup_{i=1}^{I_k} D_{i,k} \;\; \text{and} \;\; D_{i,k} \cap D_{j,k} = \emptyset, \forall i \neq j, \qquad (12)$$

and

$$\lim_{k\to\infty} \pi_k = 0. \qquad (13)$$

Here, $|\pi_k|$ represents the magnitude of π_k defined by $|\pi_k| := \max_i \operatorname{diam}(D_{i,k})$, where for a set D, $\operatorname{diam}(D) := \sup_{x,y\in D} \|x - y\|_2$ is its "diameter." Here, $\|\cdot\|_2$ is the Euclidean norm. Intuitively, condition (13) is to force the shrinkage of $D_{i,k}$ in "diameter." Note that such a partition always exists because $\mathscr{X}$ is bounded. Furthermore, the choice of the partition affects whether the resultant finite representation of $\mathscr{D}^*$ is evenly spread or not; if the partition is balanced in terms of the sizes of the partitioning sets (e.g., based on an even-grid partition), then we expect the resultant representation to be evenly spread.

The population of reference distributions $\{h_{i,k}(\cdot)\}$ could be constructed as follows. Let

$$h_{i,k}(x) \propto \mathbb{1}\{x \in D_{i,k}\}, \forall i = 1, \ldots, I_k. \qquad (14)$$

That is, $h_{i,k}(\cdot)$ is the uniform distribution on $D_{i,k}$. Note that the construction of $h_{i,k}(\cdot)$ follows the general principle from the CE method; in particular, $h_{i,k}(\cdot)$ more and more concentrates on the elite solutions as k goes to infinity because γ_k decreases, and, hence, D_k shrinks in k (see (11)). One could also construct more sophisticated reference distributions by introducing a shape function and put nonequal weights on $x \in D_{i,k}$ (see, e.g., MRAS in Hu et al. 2007).

As derived in the previous subsection, the sampling distribution parameter $\theta_{i,k}$ is then determined by

$$\theta_{i,k} = \arg\min_{\theta\in\Theta} KL(h_{i,k}(\cdot), g(\cdot;\theta)), \qquad (15)$$

which admits an analytical solution that satisfies (8).

The complete algorithm is summarized in the following Algorithm 1, which is also referred to as Algorithm "**SASMO**$_0$."

Algorithm 1 (Stochastic Adaptive Search for Multiobjective Optimization—Ideal (SASMO$_0$))

1. **Initialization**: Choose a parameterized family of densities $\{g(x;\theta) : \theta \in \Theta\}$. Specify a sequence of reference values $\{\gamma_k : k = 1, \ldots\}$ that satisfies (10).
2. **Iteration**: For the k^{th} iteration, choose a partition $\pi_k = \{D_{1,k}, \ldots, D_{I_k,k}\}$ on D_k that satisfies (12) and (13), where $D_k = \{x \in \mathscr{X} : D(x) \le \gamma_k\}$. Then construct the population of reference distributions $\{h_{i,k}(\cdot)\}$ according to (14) and determine the sampling distribution parameters $\{\theta_{i,k}\}$ by (15).
3. **Termination**: Check whether some stopping criterion is satisfied. If yes, stop and return the means of the currents sampling distributions; else, set $k = k + 1$ and go back to step 2.

Let us analyze the convergence properties of Algorithm **SASMO**$_0$. In particular, we will show that the convergence of the sampling distributions $\{g(x;\theta_{i,k}), k = 1, \ldots\}$ under appropriate assumptions, as summarized in the following theorem.

Theorem 1. *Suppose $\{g(x;\theta) : \theta \in \Theta\}$ belongs to the exponential family of densities. Further suppose that the sequence of reference values $\{\gamma_k \in (0,1] : k = 1, \ldots\}$ satisfies condition (10) and the sequence of partitions $\{\boldsymbol{\pi}_k : k = 1, \ldots\}$ satisfies conditions (12) and (13). Then $\forall\, x^* \in \mathscr{D}^*$, there exists one component from the mixture sampling distribution $\mathbf{g}_k(\cdot)$ at every iteration, denoted by $g(x;\theta_{i_k,k})$, such that the resultant sequence of distributions $\{g(x;\theta_{i_k,k}) : k = 1, \ldots\}$ satisfies*

$$\lim_{k\to\infty} \mathbb{E}_{\theta_{i_k,k}}[\Gamma(X)] = \Gamma(x^*). \qquad (16)$$

Proof of Theorem 1. Notice that $\forall\, x^* \in \mathscr{D}^*$, there exists a sequence of partitioning sets $\{D_{i_k,k} : k = 1, \ldots\}$ s.t. $x^* \in D_{i_k,k}$ and $D_{i_k,k} \in \pi_k$ because the sets in each partition π_k completely cover D_k and hence $\mathscr{D}^*$. Furthermore, by the property of the exponential family of densities in Equation (8), we have

$$\mathbb{E}_{h_{i_k,k}}[\Gamma(X)] = \mathbb{E}_{\theta_{i_k,k}}[\Gamma(X)],$$

where recall that $h_{i_k,k} \propto \mathbb{1}\{x \in D_{i_k,k}\}$.

Therefore, to show (3.11), it remains to show

$$\lim_{k\to\infty} \mathbb{E}_{h_{i_k,k}}[\Gamma(X)] = \Gamma(x^*), \text{ or equivalently,}$$
$$\lim_{k\to\infty} \mathbb{E}_{h_{i_k,k}}[\Gamma_j(X) - \Gamma_j(x^*)] = 0, j = 1, \dots, d_\theta.$$

That is, to show

$$\lim_{k\to\infty} \int_{D_{i_k,k}} \big(\Gamma_j(x) - \Gamma_j(x^*)\big) h_{i_k,k}(x)dx = 0, j = 1, \dots, d_\theta. \tag{17}$$

Given that $\Gamma(\cdot)$ is continuous on $\mathscr{X}$, we have that $\forall\ \varepsilon > 0$, $\exists \delta > 0$ s.t.

$$\|\Gamma(x) - \Gamma(x^*)\|_\infty \le \varepsilon, \ \ \forall\ x \in B_\delta(x^*),$$

where $B_\delta(x^*) := \{x \in \mathscr{X} : \|x - x^*\|_\infty \le \delta\}$ represents the neighborhood ball centered on x^* with radius δ. Further note that $\lim_{k\to\infty} |\pi_k| = 0$, we have

$$\lim_{k\to\infty} \operatorname{diam}\big(D_{i_k,k}\big) = 0.$$

Therefore, there exists a large integer K_ε depending on ε such that for all $k \ge K_\varepsilon$, $D_{i_k,k} \subseteq B_\delta(x^*)$. It follows that for all $k \ge K_\varepsilon$,

$$\int_{D_{i_k,k}} \big(\Gamma_j(x) - \Gamma_j(x^*)\big) h_{i_k,k}(x)dx \le \varepsilon \int_{D_{i_k,k}} h_{i_k,k}(x)dx = \varepsilon,$$
$$j = 1, \dots, d_\theta.$$

Therefore, (17), and, hence, Theorem 1 holds. □

Theorem 1 implies that for any solution x^* with domination measure of zero, there exists a converging sequence of sampling distributions, which consists of one component of the mixture sampling distribution from each iteration. Convergence occurs in the sense that the expectation (w.r.t. this sequence of sampling distributions) of the sufficient statistic converge to its function value at the solution x^*. In particular, for the multivariate normal family, the mean vectors of the normal distributions converge to the optimal solution, and the variance matrices converge to a zero matrix (which is shown by corollary 1 in Hu et al. 2007). In other words, the normal sampling distributions converge to a degenerate distribution on an optimal solution. In all of our numerical experiments (see Section 4), we use the multivariate normal family as the sampling distribution.

In practice, Algorithm **SASMO**$_0$ is not "implementable" for the following reasons: (1) the set D_k, which is regarded as the promising region of the solution space, could not be explicitly constructed, because the domination measure $D(x)$ is unknown, and (2) solving for the sampling parameters analytically without sampling through minimizing KL divergence is unlikely because the reference sampling distributions $h_{i,k}(\cdot)$ do not have an explicit characterization. In light of these challenges, we propose the following implementable version of **SASMO**$_0$.

3.2. An Implementable Version Algorithm

To have an implementable algorithm, let us introduce a sampling step at each iteration, in which multiple i.i.d. candidate solutions are drawn according to the mixture sampling distribution. The motivations are as follows.

First, the domination measure $D(x)$ could be estimated, which will then be used to determine the reference value γ_k and characterize the set of promising solutions D_k at the kth iteration. In particular, suppose N_k i.i.d. candidate solutions $\{x_k^1, \dots, x_k^{N_k}\}$ are drawn according to the mixture sampling distribution $\mathbf{g}_k(\cdot)$, then the domination measure for x_k^i could be estimated by

$$\widetilde{D}(x_k^i) = \frac{1}{N_k \cdot \nu(\mathscr{X})} \sum_{j=1}^{N_k} \frac{1}{\mathbf{g}_k\left(x_k^j\right)} \mathbb{1}\left\{x_k^j \prec_d x_k^i\right\} \tag{18}$$

via the principle of importance sampling by changing the underlying probability measure of the expectation $D(x)$ from $\mathscr{U}(\cdot)$ to $\mathbf{g}_k(\cdot)$.

Second, the partition $\boldsymbol{\pi}_k$ could be determined based on the evaluations of candidate solutions and a clustering algorithm. In particular, the partitioning sets $\{D_{i_k,k}\}$ could be characterized by resultant clusters and $|\boldsymbol{\pi}_k|$ could be approximated by radii of the clusters. Then the reference distribution $h_{i_k,k}(\cdot)$ could be characterized by an empirical (uniform) distribution on the candidate solutions in the corresponding cluster.

Third, the sampling parameter $\theta_{i_k,k}$ could be solved by minimizing the KL divergence between the empirical reference distribution and $g(x; \theta_{i_k,k})$. We formally describe Algorithm "**SASMO**$_1$"— an implementable version of algorithm **SASMO**$_0$, in the following Algorithm 2.

Algorithm 2 (Stochastic Adaptive Search for Multiobjective Optimization—Implementable (SASMO$_1$))

1. **Initialization**: Choose a parameterized family of densities $\{g(x; \theta) : \theta \in \Theta\}$ with initial parameter $\theta_{1,0}$ with $\widetilde{I}_0 = 1$. Specify a mixing coefficient $\alpha \in (0, 1)$, a percent quantile ρ, a sample size sequence $\{N_k\}$. Set $k = 0$.
2. **Sampling**: Draw N_k i.i.d. candidate solutions $\{x_k^i : i = 1, 2, \dots, N_k\}$ according to $\mathbf{g}_k(\cdot)$, where
$$\mathbf{g}_k(\cdot) \triangleq (1 - \alpha) g_k(\cdot) + \alpha \mathscr{U}_{\mathscr{X}}(\cdot) \tag{19}$$
where recall that $g_k(x) = \frac{1}{\tilde{I}_k} \sum_{i=1}^{\tilde{I}_k} g(x; \theta_{i,k})$, and $\mathscr{U}_{\mathscr{X}}(\cdot)$ represents the uniform distribution on $\mathscr{X}$.
3. **Estimation**: For $i = 1, \dots, N_k$, estimate the domination measure $D(x_k^i)$ at x_k^i by (18). Sort the estimates $\{\widetilde{D}(x_k^i)\}$ in ascending order, denoted by $\widetilde{D}(x_k^{(1)}) \le \widetilde{D}(x_k^{(2)}) \le \cdots \le \widetilde{D}(x_k^{(N_k)})$. Set the

reference value $\widetilde{\gamma}_k$ to be the sample ρ-percent quantile $\widetilde{D}(x_k^{(\lceil \rho N_k \rceil)})$, that is, $\widetilde{\gamma}_k = \widetilde{D}(x_k^{(\lceil \rho N_k \rceil)})$, where $\lceil \rho N_k \rceil$ is the smallest integer that is greater than or equal to ρN_k.

4. **Updating**: Construct the set of elite candidate solutions $\widetilde{D}_k$ by $\widetilde{D}_k := \{x_k^i : \widetilde{D}(x_k^i) \le \widetilde{\gamma}_k\}$. Use a clustering algorithm (e.g., Algorithm 3) to determine the number of clusters $\widetilde{I}_{k+1}$ and cluster the elite solution set $\widetilde{D}_k$ into partition $\widetilde{\boldsymbol{\pi}}_k := \{\widetilde{D}_{1,k}, \ldots, \widetilde{D}_{\widetilde{I}_{k+1},k}\}$. Compute the sampling parameter $\theta_{i,k+1}$ based on the set of elite candidate solutions $\widetilde{D}_{i,k}$ by solving

$$\theta_{i,k+1} \overset{\Delta}{=} \arg\max_{\theta\in\Theta} \frac{1}{|\widetilde{D}_{i,k}|} \sum_{x\in\widetilde{D}_{i,k}} \frac{\ln g(x;\theta)}{\mathbf{g}_k(x)}, i = 1, \ldots, \widetilde{I}_{k+1}. \tag{20}$$

5. **Stopping**. Check whether some stopping criterion is satisfied. If yes, stop and return the means of the components of $\mathbf{g}_{k+1}(\cdot)$; else, set $k = k + 1$ and go back to step 2.

In the initialization step (step 1), a common choice of $\{g(x;\theta)\}$ is an exponential family of densities. The initial sampling parameter $\theta_{1,0}$ should be chosen in a way such that the resultant sampling distribution is close to the uniform distribution on $\mathscr{X}$, so that the entire solution space is evenly explored in the early iterations. For example, if $\{g(x;\theta)\}$ is the family of multivariate Gaussian distributions, then $\theta_{1,0}$ is characterized by the mean vector $\mu_{1,0}$ and covariance matrix $\Sigma_{1,0}$. To enforce global exploration of the entire solution space, $\Sigma_{1,0}$ needs to be relatively large. Furthermore, as we will see later, the mixture coefficient α, the percent quantile ρ, and the sample size sequence $\{N_k\}$ also affect the robustness and convergence of the algorithm.

In the sampling step (step 2), note that in (19) the sampling distribution $\mathbf{g}_k(\cdot)$ is a combination of the uniform distribution on $\mathscr{X}$ and the mixture parameterized sampling distribution obtained from previous iteration. Having the uniform distribution component helps maintain a global exploration of the solution space; the greater α is, the more emphasis is on the global exploration. A typical choice of α is $\alpha = 0.1$. Another benefit is that it also helps control the variances for the estimators of domination measure; in particular, the variances are uniformly bounded if $\{N_k\}$ is bounded from below, which is generally required in a model-based algorithm.

In the estimation step (step 3), for the sake of convergence, the sample size sequence $\{N_k\}$ is either set to be a large constant or strictly increasing by τ_k, that is, $N_{k+1} = \tau_k N_0$ such that $\tau_k > 1$ for all k. The quantile level ρ controls the magnitude of the reference value $\widetilde{\gamma}_k$; that is, it determines the percentage of solutions that are being treated as elite solutions and used to update the sampling distribution for the next iteration. Therefore, it also determines the trade-off between the exploitation of the promising regions and the exploration of the entire solution space. When a smaller ρ is used, fewer elite solutions are used in the updating the sampling distribution, which leads to one that concentrates more on the neighborhoods of the elite solutions, that is, the promising regions.

In the updating step (step 4), the set of elite solutions $\widetilde{D}_k$ characterizes the promising region D_k, and the set of clusters $\widetilde{\boldsymbol{\pi}}_k$ characterizes the partition $\boldsymbol{\pi}_k$. The selection of clustering algorithm is essential because the resultant $\widetilde{\boldsymbol{\pi}}_k$ needs to satisfy

$$\lim_{k\to\infty} |\widetilde{\pi}_k| = 0, \text{ where } |\widetilde{\pi}_k| = \max_{1\le i\le \widetilde{I}_k} \operatorname{diam}\left(\widetilde{A}_{i,k}\right).$$

Furthermore, to have a good representation of $\mathscr{D}^*$, the clusters are preferred to be evenly spread. Following the guideline, a specific clustering algorithm (Algorithm 3) will be described in detail later.

In the termination step (step 5), a common stopping criterion is when the threshold distance in the clustering algorithm falls below a prespecified limit (if a threshold-based clustering algorithm is used). The means of the components in the final mixture sampling distribution form a finite representation of $\mathscr{D}^*$. Furthermore, the number of solutions in the representation of $\mathscr{D}^*$ could be determined by the threshold limit in the clustering algorithm. To provide more clarity of the proposed algorithm, we provide an illustration of the evolution of $\mathbf{SASMO_1}$ in Figure 3 (note that the problem is taken from Example 2, but the sampling distributions in each iteration are illustrative and not taken from actually running the algorithm).

The convergence of $\mathbf{SASMO_1}$ is difficult to analyze because of multiple layers of approximations and the resultant intertwined errors from sampling, estimating the domination measure from the samples, and clustering the elite samples. Nevertheless, the algorithm $\mathbf{SASMO_1}$ works intuitively as a practical extension of $\mathbf{SASMO_0}$. We will show that it performs competitively to or outperforms some of the existing algorithms in numerical experiments.

We conclude this section by introducing the threshold-based clustering algorithm (adapted from Bhatia 2004) used in step 4 of $\mathbf{SASMO_1}$, as described in the following Algorithm 3.

Algorithm 3 (A Threshold-Based Clustering Algorithm)

Input: Threshold distance Δ_k, elite solution set $\widetilde{D}_k$, and a shrinking factor $C > 1$.

Output: The set of clusters $\widetilde{\boldsymbol{\pi}}_k$ and threshold distance Δ_{k+1}.

1. **Initialization**: Randomly select a solution from $\widetilde{D}_k$. This solution is defined as the centroid of cluster $\widetilde{D}_{1,k}$. Set $\widetilde{I}_{k+1} = 1$.

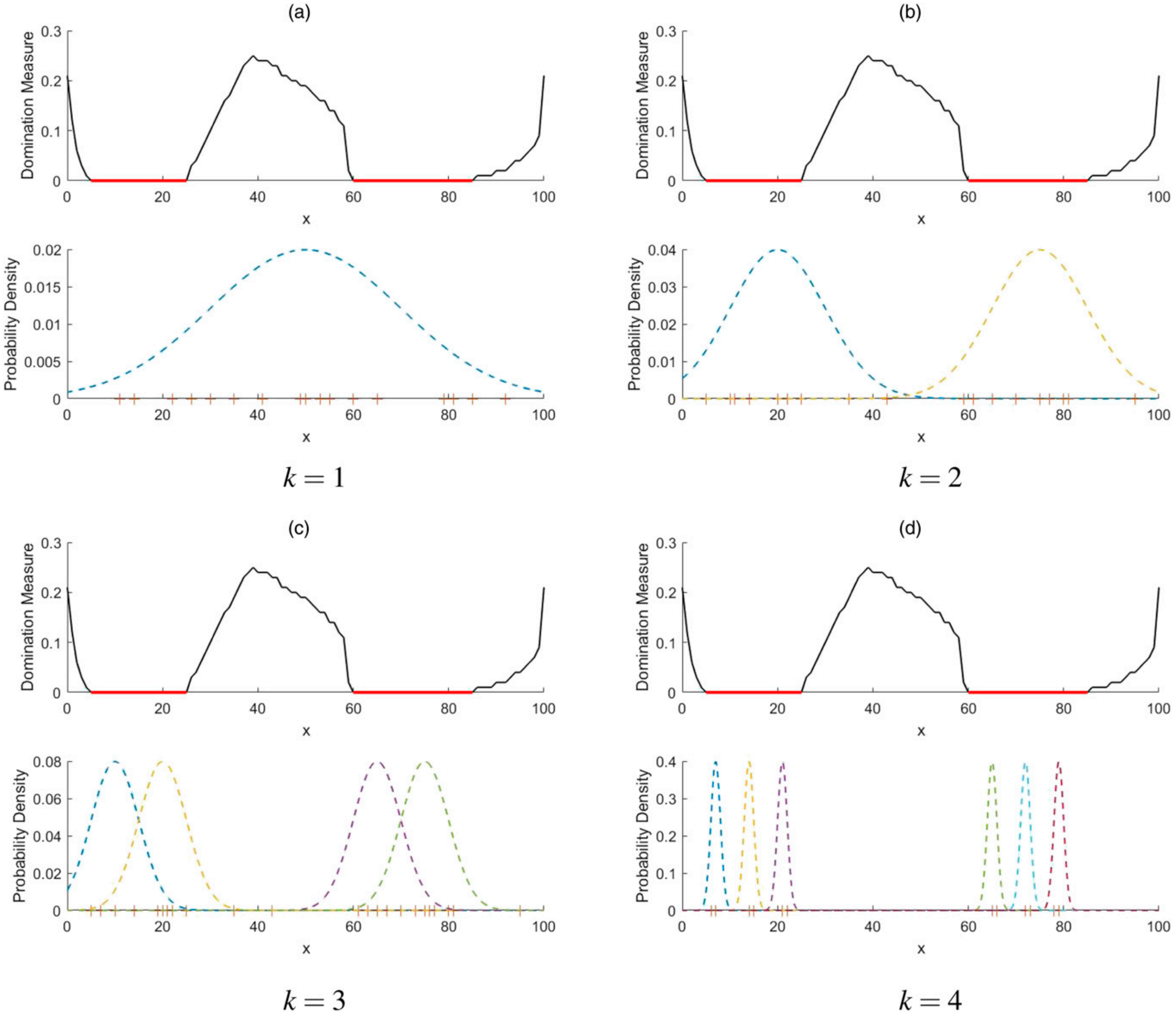
2. **Iteration**: Randomly select a solution from $\widetilde{D}_k$ that has not been assigned. Compute the Euclidean distances from that solution to the centroids of existing clusters in a randomized order. Assign the solution to the first cluster found with distance less than Δ_k, and update the centroid of that cluster as the average of the solutions in the cluster. If no such cluster is found, create a new cluster where the solution is the centroid of the new cluster, and set $\widetilde{I}_{k+1} = \widetilde{I}_{k+1} + 1$.
3. **Termination**: Check whether any solution from $\widetilde{D}_k$ has not been assigned. If yes, go to step 2; otherwise, return the set of clusters $\widetilde{\boldsymbol{\pi}}_k$ and the threshold distance at next iteration by

$$\Delta_{k+1} \triangleq \min\left[\frac{1}{C\cdot\widetilde{I}_{k+1}}\sum_{i=1}^{\widetilde{I}_{k+1}} Tr\left(\widetilde{\Sigma}_{i,k}\right), \frac{\Delta_k}{C}\right], \qquad (21)$$

where $\widetilde{\Sigma}_{i,k}$ is the sample variance of cluster $\widetilde{D}_{i,k}$, and $Tr(\cdot)$ is the trace of a matrix.

In the iteration step (step 2) of Algorithm 3, we randomize the order in which the distances from the selected solution and the centroids of the existing clusters are compared. This is to prevent one cluster from becoming significantly larger than others. In the termination step (step 3), the threshold distance at the next iteration Δ_{k+1} is decreasing adaptively, noting that $1/\widetilde{I}_{k+1}\cdot\sum_{i=1}^{\widetilde{I}_{k+1}} Tr(\widetilde{\Sigma}_{i,k})$ is an empirical measure of how solutions within each cluster are close to each other. If this measure is still large, then the threshold distance is forced to decrease by at least a factor of C. Therefore, the shrinking factor C determines how fast Algorithm **SASMO₁** terminates and how many solutions are generated in the resultant representation of $\mathscr{D}^*$.

Figure 3. (Color online) Illustration of the Evolution of **SASMO$_1$**



$k = 1$

$k = 2$

$k = 3$

$k = 4$

4. Numerical Experiments

We will demonstrate by numerical results that Algorithm $\mathbf{SASMO_1}$ is (1) not sensitive to the geometry of the Pareto optimal set or the Pareto front, (2) scalable in terms of decision variables and objective functions, and (3) competitive with existing methods in terms of how close the solutions are to the true Pareto optimal set and how evenly spread the solutions are in the solution space. To this end, we evaluate the performance of Algorithm $\mathbf{SASMO_1}$ on test functions from the ZDT (Deb 2001), DTLZ (Deb 2001), and Van Veldhuizen's (Coello et al. 2002) test suites (the functions are listed in the online supplement) and compare our results with the following existing methods:

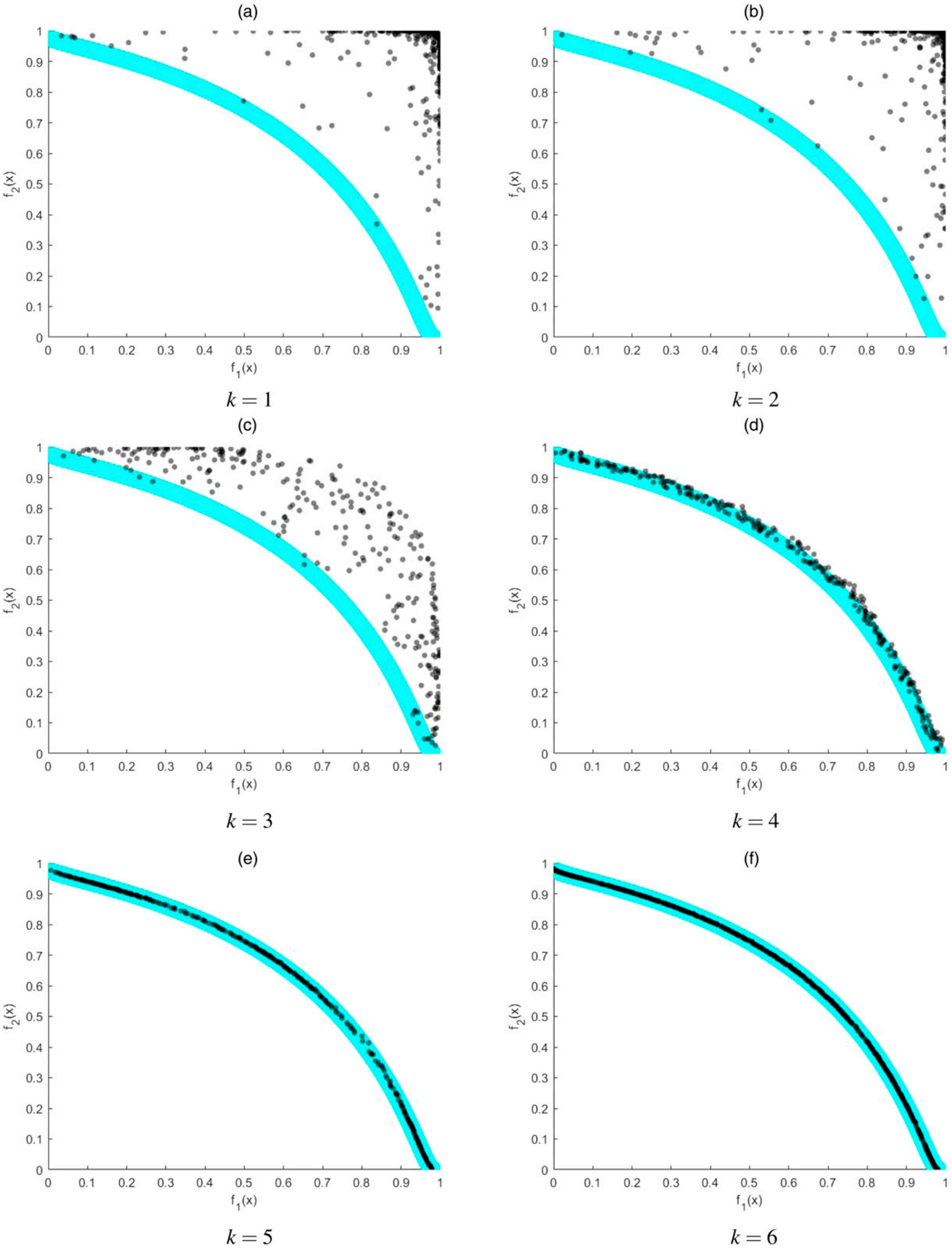
- Elitist nondominated sorting genetic algorithm (NSGA II) (Deb 2001)
- Strength Pareto evolutionary algorithm (SPEA-II) (Kim et al. 2004)
- Pareto envelope-based selection algorithm (PSEA-II) (Corne et al. 2001)
- Multiobjective particle swarm optimization (MOPSO) (Coello et al. 2002)

The above methods are all evolutionary algorithms and differ in how the population of candidate solutions are selected and maintained. In NSGA-II the population of solutions is divided based on the following rule: the first group of solutions is all nondominated and the second group of solutions is only dominated by the solutions in the first group. The grouping is continued in this manner until all solutions are classified. Once the solutions are divided, a crowding distance is calculated, which measures how close a solution is to its neighbors. Solutions are selected based on their group classification and crowding distance, and new solutions are generated from crossover and mutation operators. SPEA-II is an extended version of the original SPEA algorithm (Zitzler and Thiele 1998) that includes a specialized ranking system to order the solutions based on their fitness values, which is an objective function that summarizes how close a given solution is to the Pareto front. SPEA-II keeps an archive of all solutions generated starting from the initialization of the algorithm and constructs a population of solutions by combining the archived solutions with the solutions generated at the current iteration. All nondominated solutions in the population are assigned fitness values such that the search is directed toward the true Pareto front. PSEA-II introduces a new selection technique where the objective space is divided into hyperboxes and solutions are randomly selected from those hyperboxes. The fitness value of a nondominated solution depends on the number of nondominated solutions that occupy that same hyperbox. This method of selection is shown to result in a good spread of solutions in the objective space. MOPSO is a particle swarm method that includes a constraint-handling mechanism and a mutation operator that substantially improves the exploration ability of the original algorithm.

The instances in the ZDT test suite have a scalable number of parameters. Therefore, this test suite tests the ability of an algorithm to converge to the Pareto front and obtain diverse solutions in a high-dimensional solution space. The challenge in dealing with a high dimensional solution space is that it is more difficult to get an evenly spread set of solutions in the solution space. Moreover, this test suite includes test function ZDT4 that has 21^9 local Pareto optimal fronts. Functions similar to ZDT4 may cause premature convergence to local optimal regions of the solution space. In practice the structure of the Pareto front is unknown; therefore, it is important to see how our method performs with different Pareto front geometries. Consequently, we consider the test function from the Van Veldhuizen's test suite, which offers a variety of Pareto front geometries. The different structures of the Pareto front can be convex, concave, degenerate, mixed, continuous, discontinuous, or contain flat regions. In this context, flat regions are areas of the objective space, where relatively small perturbations of parameters in the solution space do not affect the objective values. Each geometric structure presents its own difficulty. For instance, problems with isolated points are difficult to solve because information in the surrounding region usually does not indicate whether a Pareto optimal solution is nearby. Furthermore, a function that has a many to one mapping between the solution space and the objective space are difficult to solve because of the flat regions in the objective space. The last set of problems that we consider is the DLTZ test suite, which has a scalable number of objectives while also having complicated Pareto front geometries. The increase in dimensionality of the objective space causes difficulty with selecting the best solutions. A large number of objectives causes a majority of solutions to be nondominated by each other, which may throttle an algorithm's convergence to the true Pareto front. The problems chosen from the aforementioned test suites are included in the online supplement, and more details on the problems properties and challenges can be found in Huband et al. (2006).

We solve all problems with Algorithm $\mathbf{SASMO_1}$, which terminates when the threshold distance falls below the threshold bound $\bar{\Delta}$ or when 10,000 objective function evaluations are executed. We choose the following parameter values and also include recommended parameter ranges for the tested problems: initial sample size $N_0 = 300$ ($300 \le N_0 \le 600$), $\tau_k = 1.01$ ($1 < \tau_k \le 1.05$), $N_k = \tau_k^k N_0$, percent quantile $\rho = 0.1$ ($.05 \le \rho \le .20$), mixing coefficient $\alpha = 0.1$ ($.1 \le \alpha \le .2$), threshold bound $\bar{\Delta} = 0.001$ ($.001 \le \bar{\Delta} \le .1$), threshold distance shrinking factor $C = 1.1$, initial mean $\mu_0 = \mathbf{0}$, and initial covariance matrix $\Sigma_0 = 1000\mathbf{I}_d$. To provide more insight on how Algorithm $\mathbf{SASMO_1}$ converges to the Pareto front, we provide an illustration in Figure 4. Each subfigure in Figure 4 corresponds to one iteration of the algorithm. The black circles represent the elite solutions generated by the sampling distribution in that iteration, and the cyan thick curve represents the true Pareto front of test function ZDT2.

Figure 4. (Color online) Convergence of Algorithm $\mathbf{SASMO_1}$ on ZDT2



$k = 1$

$k = 2$

$k = 3$

$k = 4$

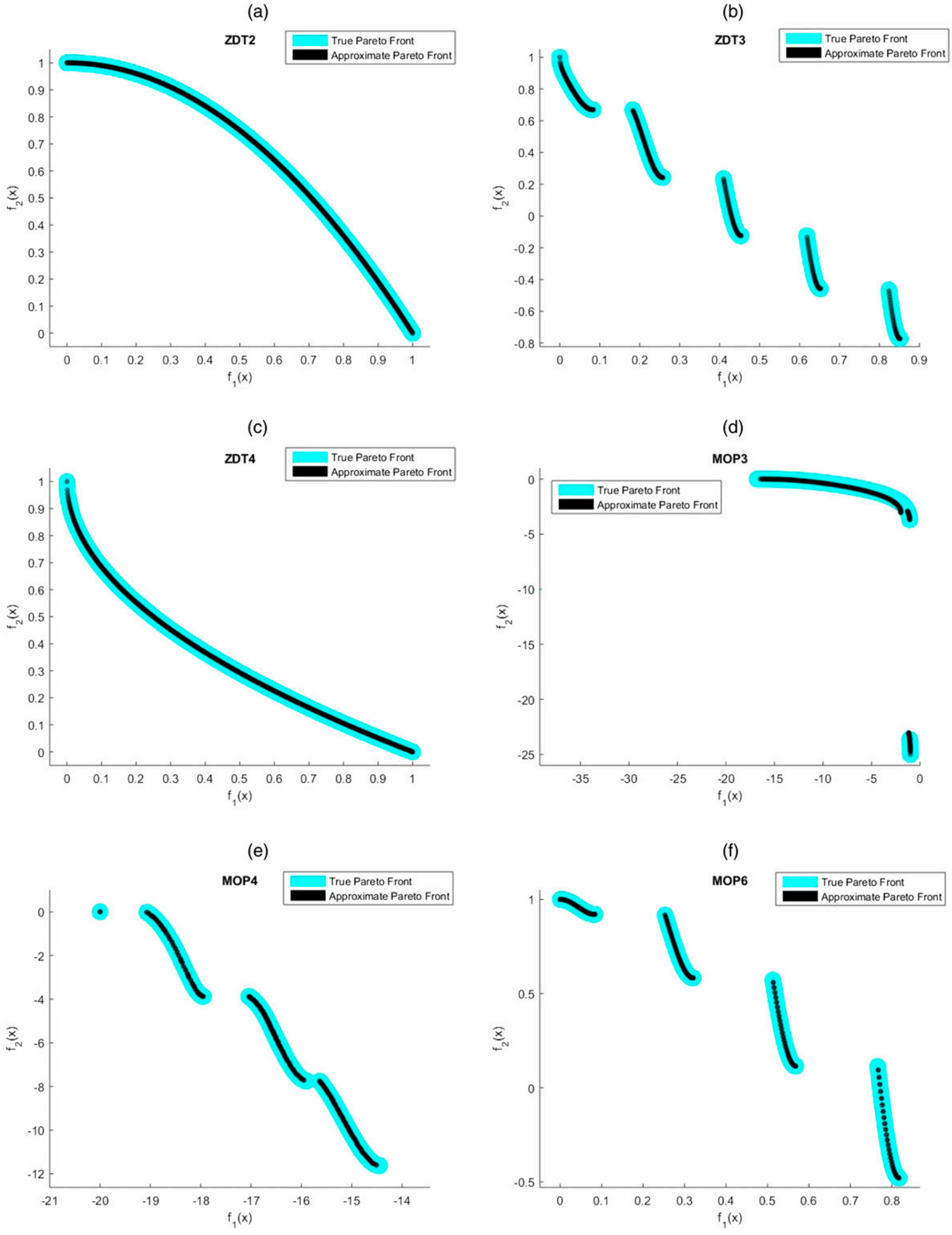
$k = 5$

$k = 6$

We also include such a figure in the online supplement to show how **SASMO₁** converges on the MOP4 test function, which has a discontinuous Pareto front.

The resultant approximations of the Pareto fronts for the tested problems are illustrated in Figure 5. Note that for each subfigure of Figure 5, the black thin curve represents the approximated Pareto front produced by Algorithm **SASMO₁** in the objective space (f_1, f_2). For the tested cases, we observe that Algorithm **SASMO₁** is capable of obtaining isolated Pareto optimal points and capturing the entire Pareto front for problems that have multiple discontinuous Pareto curves. These results also demonstrate that relaxing the concept of Pareto dominance to domination

Figure 5. (Color online) Approximate Pareto Front vs. True Pareto Front



measure does not affect the solution quality of our algorithm.

In order to quantify the performance of Algorithm **SASMO$_1$** and compare it with the performances of the aforementioned existing algorithms, we use the convergence metric Λ and diversity metric Υ defined in Deb (2001).

The convergence metric Λ in the objective space is defined by

$$\Lambda \triangleq \frac{1}{|R|}\sum_{y:\,\mathbf{f}(y)\in R}\left\{\min_{x\in Z}\|\mathbf{f}(x)-\mathbf{f}(y)\|_2\right\},$$

where R is a prespecified reference set consisting of $|R|$ uniformly spread points from the true Pareto front and we choose $|R| = 500$, $\mathbf{f}(\cdot) = (f_1(\cdot),\ldots,f_n(\cdot))$ is the vector of objective functions, and Z is the set of approximate Pareto front generated by the algorithm of interest. In other words, the convergence metric Λ can be regarded as the average distance from all points in the reference set to the approximate Pareto front, which measures the closeness of the approximate Pareto front to the true Pareto front. Therefore, the smaller the value is for Λ, the closer the approximate Pareto front is to the true Pareto front.

Before introducing the diversity metric Υ in the solution space, let us first order the obtained $|Z|$ approximate Pareto optimal solutions $\{x^1,\ldots,x^{|Z|}\}$ generated from an algorithm of interest by $x_1^{(1)} \le \cdots \le x_1^{(|Z|)}$. That is, they are ordered by the values of their first components. We also let $x^{(1)}$ and $x^{(|Z|)}$ be the left and right boundary points of the approximate Pareto optimal set, and let x^l and x^r be the left and right boundary of the true Pareto optimal set also in terms of the value of a solution's first component. The diversity metric Υ is defined by

$$\Upsilon = \frac{d_l + d_r + \sum_{i=1}^{|Z|-1}\left|d_i - \bar{d}\right|}{d_l + d_r + (|Z|-1)\bar{d}},$$

where $d_l := \|x^l - x^{(1)}\|$ ($d_r := \|x^{(r)} - x^{(|Z|)}\|$) is the distance between the left (right) boundary point of the true Pareto optimal set and the left (right) boundary point of the approximate Pareto optimal set, $d_i := \|x^{(i+1)} - x^{(i)}\|$ is the distance between an approximate Pareto optimal solution and its closest neighbor, and $\bar{d}$:=1/(|Z|−1) $\sum_{i=1}^{|Z|-1} d_i$ is the average of these distances. Essentially, Υ measures how well the solutions are evenly spaced in the solution space. The smaller the value is for this metric, the closer the approximate Pareto optimal set is from being uniformly distributed.

Table 1. Comparisons of the Convergence Metric Λ

Problem	**SASMO$_1$**	NSGA II	SPEA II	MOPSO	PSEA II
	Λ	Λ	Λ	Λ	Λ
ZDT2	**0.0051**	0.1082	0.2317	0.4453	0.3214
ZDT3	**0.1096**	0.2265	0.4955	0.5724	0.6204
ZDT4	**0.0144**	0.02432	0.05756	0.0321	0.1233
MOP3	**0.0358**	0.0495	0.1278	1.0287	1.3432
MOP4	0.2311	0.1261	0.3779	0.5615	0.4572
MOP5	**0.0337**	0.0492	0.0357	0.0614	0.1738
MOP6	**0.0406**	0.0752	0.1618	1.6583	1.1164
DTLZ1	3.4998	3.0620	3.1588	5.0017	4.5013
DTLZ2	**3.8709**	4.2508	5.6672	6.8525	6.1093

Note. When **SASMO$_1$** outperforms all other algorithms on a problem, its convergence metric value is in bold.

For each problem instance, we perform 30 independent replications of each method implemented in MATLAB. The codes for the existing methods can be found at http://yarpiz.com/category/multiobjective-optimization. We report the average values for the convergence metric Λ and the diversity metric Υ obtained from the 30 trials for each method. For the existing methods, the parameters are chosen so that the maximum computational budget is 10,000 objective evaluations for all algorithms tested. Therefore, we choose the following parameters:

- Number of generations: 100
- Population size: 100
- Archive size: 1,000.

The results are summarized in Tables 1 and 2.

We can see that Algorithm **SASMO$_1$** outperforms all tested existing methods on 7 out of the 9 problems in respect to both the convergence and the diversity metrics, and on the remaining three problems **SASMO$_1$** outperforms in one metric and performs competitively in the other metric. The favorable results w.r.t. the diversity metric is likely because the center of each cluster represents an estimated Pareto optimal solution, and it is at least the threshold distance away from the closest cluster. As a result, the distance between each estimated Pareto optimal solution is close to the threshold bound for most of the problems. Another takeaway from the numerical results is how well our algorithm performed on problems that had discontinuous Pareto fronts. This is likely

Table 2. Comparisons of the Diversity Metric Υ

Problem	**SASMO$_1$**	NSGA II	SPEA II	MOPSO	PSEA II
	Υ	Υ	Υ	Υ	Υ
ZDT2	**0.5910**	0.8713	1.1884	2.0140	1.4732
ZDT3	**0.7215**	0.8991	1.5246	1.0231	0.9353
ZDT4	**0.3011**	0.3447	0.6361	1.8243	1.1003
MOP3	**0.0223**	0.4395	1.0175	0.9254	1.6739
MOP4	**0.0397**	0.3466	0.7324	0.7551	0.8482
MOP5	**0.0253**	0.4121	0.8522	1.3377	0.8986
MOP6	**0.0195**	0.4112	0.7352	0.9037	0.5830
DTLZ1	1.8721	2.5336	2.9472	3.5023	1.0642
DTLZ2	1.2454	1.0018	1.1741	2.0282	2.0012

Note. When **SASMO$_1$** outperforms all other algorithms on a problem, its diversity metric value is in bold.

Table 3. Total Number of Function Evaluations When **SASMO$_1$** Terminates

ZDT2	ZDT3	ZDT4	MOP3	MOP4	MOP5	MOP6	DTLZ1	DTL2
2,362	9,231	9,867	6,455	7,185	10,000	9,621	10,000	10,000

because an adaptive number of components is used in the mixture sampling distribution so that each promising region of the solution space is thoroughly explored.

Because **SASMO$_1$** terminates when the threshold distance is below the lower bound, the total number of objective function evaluations indicates how fast the algorithm converges. Table 3 shows the total number of objective evaluations (averaged over 30 trials) when **SASMO$_1$** terminates for each test problem. Problems with more than two objectives used the maximum budget at each trial because of the difficulty of identifying nondominated solutions. Furthermore, the number of objectives required until convergence increases as the complexity of the Pareto front increases. In conclusion, we show empirically that Algorithm **SASMO$_1$** gives satisfactory results regardless of the geometry of the Pareto front and is competitive with several existing algorithms.

5. Conclusions and Future Research

In this paper, we introduce a novel performance metric called domination measure to measure the quality of a solution in a multiobjective problem. The concept of domination measure transforms the original problem to a stochastic single-objective problem with a softened goal of finding optimal solutions that have a domination measure of zero. This opens the door to many new algorithms, particularly by adapting existing stochastic optimization algorithms. Here we propose a model-based approach to find a finite and approximately uniformly spread representation of the set of solutions with domination measure of zero, which is close to a finite and approximately uniformly spread representation of the Pareto optimal set. We present an ideal algorithm that has nice convergence properties, and its implementable version that has competitive numerical performances compared with many existing approaches. More sophisticated approaches and algorithms based on domination measure can be incorporated to further improve the results on theoretical convergence and numerical performance.

References

Auger A, Bader J, Brockhoff D, Zitzler E (2012) Hypervolume-based multiobjective optimization: theoretical foundations and practical implications. *Theoret. Comput. Sci.* 425(March): 75–103.

Bader J, Zitzler E (2011) Hype: an algorithm for fast hypervolume-based many-objective optimization. *Evol. Comput.* 19(1):45–76.

Bader J, Deb K, Zitzler E (2010) Faster hypervolume-based search using Monte Carlo sampling. *Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems* (Springer, New York), 313–326.

Bekker J, Aldrich C (2011) The cross-entropy method in multi-objective optimisation: An assessment. *Eur. J. Oper. Res.* 211(1):112–121.

Beume N, Naujoks B, Emmerich M (2007) Sms-emoa: Multiobjective selection based on dominated hypervolume. *Eur. J. Oper. Res.* 181(3):1653–1669.

Bhatia SK (2004) Adaptive k-means clustering. Barr V, Markov Z, eds. *Proc. 17th Internat. Florida Artificial Intelligence Res. Soc. Conf.* (AAAI Press, Menlo Park, CA), 695–699.

Coello CAC, Van Veldhuizen DA, Lamont GB (2002) *Evolutionary Algorithms for Solving Multi-Objective Problems*, vol. 242 (Springer, New York).

Corne DW, Jerram NR, Knowles JD, Oates MJ (2001) Pesa-ii: region-based selection in evolutionary multiobjective optimization. Aguirre H, ed. *Proc. Genetic Evolutionary Comput. Conf. (GECCO'2001)* (ACM, New York).

Custódio AL, Madeira JA, Vaz AIF, Vicente LN (2011) Direct multisearch for multiobjective optimization. *SIAM J. Optim.* 21(3): 1109–1140.

Deb K (2001) *Multi-Objective Optimization Using Evolutionary Algorithms*, vol. 16 (John Wiley & Sons, Hoboken, NJ).

Feldman G, Hunter SR, Pasupathy R (2015) Multi-objective simulation optimization on finite sets: optimal allocation via scalarization. *2015 Winter Simulation Conf. (WSC)* (IEEE Press, Piscataway, NJ), 3610–3621.

Fleischer M (2003) The measure of Pareto optima applications to multi-objective metaheuristics. *Evolutionary Multi-Criterion Optimization* (Springer, New York), 519–533.

Hu J, Fu MC, Marcus SI (2007) A model reference adaptive search method for global optimization. *Oper. Res.* 55(3):549–568.

Huband S, Hingston P, Barone L, While L (2006) A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans. Evolutionary Comput.* 10(5):477–506.

Kim M, Hiroyasu T, Miki M, Watanabe S (2004) SPEA2+: Improving the performance of the strength Pareto evolutionary algorithm 2. *Parallel Problem Solving from Nature-PPSN VIII* (Springer, New York), 742–751.

Köksalan M, Phelps S (2007) An evolutionary metaheuristic for approximating preference-nondominated solutions. *INFORMS J. Comput.* 19(2):291–301.

Le Riche R, Saouab A, Bréard J (2003) Coupled compression rtm and composite layup optimization. *Compos. Sci. Tech.* 63(15): 2277–2287.

Lee LH, Chew EP, Teng S, Chen Y (2008) Multi-objective simulation-based evolutionary algorithm for an aircraft spare parts allocation problem. *Eur. J. Oper. Res.* 189(2):476–491.

Lee LH, Chew EP, Teng S, Goldsman D (2010) Finding the nondominated pareto set for multi-objective simulation models. *IIE Trans.* 42(9):656–674.

Mete HO, Zabinsky ZB (2014) Multiobjective interacting particle algorithm for global optimization. *INFORMS J. Comput.* 26(3): 500–513.

Minella G, Ruiz R, Ciavotta M (2008) A review and evaluation of multiobjective algorithms for the flowshop scheduling problem. *INFORMS J. Comput.* 20(3):451–471.

Molina J, Laguna M, Martí R, Caballero R (2007) Sspmo: A scatter tabu search procedure for non-linear multiobjective optimization. *INFORMS J. Comput.* 19(1):91–100.

Nam D, Park CH (2000) Multiobjective simulated annealing: A comparative study to evolutionary algorithms. *Internat. J. Fuzzy Systems* 2(2):87–97.

Pal M, Bandyopadhyay S (2016) Reliability of convergence metric and hypervolume indicator for many-objective optimization. *2016 2nd Internat. Conf. Control, Instrumentation, Energy & Comm. (CIEC)* (IEEE, Piscataway, NJ), 511–515.

Qasem SN, Shamsuddin SM (2011) Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis. *Appl. Soft Comput.* 11(1):1427–1438.

Rubinstein RY (2001) Combinatorial optimization, cross-entropy, ants and rare events. Uryasev S, Pardalos PM, eds. *Stochastic Optimization: Algorithms and Applications* (Springer, New York), 303–363.

Santiago A, Huacuja HJF, Dorronsoro B, Pecero JE, Santillan CG, Barbosa JJG, Monterrubio JCS (2014) A survey of decomposition methods for multi-objective optimization. Castillo O, Melin P, Pedrycz W, Kacprzyk J, eds. *Recent Advances on Hybrid Approaches for Designing Intelligent Systems* (Springer, New York), 453–465.

Toffolo A, Lazzaretto A (2002) Evolutionary algorithms for multi-objective energetic and economic optimization in thermal system design. *Energy* 27(6):549–567.

Unveren A, Acan A (2007) Multi-objective optimization with cross entropy method: stochastic learning with clustered pareto fronts. *2007 IEEE Congress on Evolutionary Comput.* (IEEE Press, Piscataway, NJ), 3065–3071.

Zabinsky ZB (2013) *Stochastic Adaptive Search for Global Optimization*, vol. 72. (Springer, New York).

Zhou E, Hu J (2014) Gradient-based adaptive stochastic search for non-differentiable optimization. *IEEE Trans. Automat. Control.* 59(7):1818–1832.

Zitzler E, Thiele L (1998) Multiobjective optimization using evolutionary algorithms—a comparative case study. Eiben AE, Bäck T, Schoenauer M, Schwefel H-P, eds. *Internat. Conf. Parallel Problem Solving Nature* (Springer, New York), 292–301.

Zitzler E, Brockhoff D, Thiele L (2007) The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. Trautmann H, Rudolph G, Klamroth K, Schütze O, Wiecek M, Jin Y, Grimme C, eds. *Evolutionary Multi-Criterion Optimization* (Springer, New York), 862–876.

Zitzler E, Thiele L, Laumanns M, Fonseca CM, Da Fonseca VG (2003) Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans. Evolutionary Comput.* 7(2): 117–132.