Proceedings of the 2011 Winter Simulation Conference
S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, eds.

THE SAMPLE AVERAGE APPROXIMATION METHOD FOR MULTI-OBJECTIVE STOCHASTIC OPTIMIZATION

Sujin Kim
Jong-hyun Ryu

National University of Singapore
E1A Engineering Drive 2
Singapore, 117576

ABSTRACT

In this paper, we consider black-box problems where the analytic forms of the objective functions are not available, and the values can only be estimated by output responses from computationally expensive simulations. We apply the sample average approximation method to multi-objective stochastic optimization problems and prove the convergence properties of the method under a set of fairly general regularity conditions. We develop a new algorithm, based on the trust-region method, for approximating the Pareto front of a bi-objective stochastic optimization problem. At each iteration of the proposed algorithm, a trust region is identified and quadratic approximate functions for the expected objective functions are built using sample average values. To determine non-dominated solutions in the trust region, a single-objective optimization problem is constructed based on the approximate objective functions. After updating the set of non-dominated solutions, a new trust region around the most isolated point is determined to explore areas that have not been visited. The numerical results show that our proposed method is feasible, and the performance can be significantly improved with an appropriate sample size.

1 INTRODUCTION

Multi-objective optimization arises in a wide variety of applications, whenever it is necessary to make a tradeoff between different important, but conflicting goals. Examples include portfolio optimization, where we search for a position that optimizes both return and risk (Markowitz 1991), or new product design, where we may need to simultaneously optimize several critical factors (Wilson et al. 2001, Tappeta et al. 2002, Shan and Wang 2005). The usual concept of optimality from single-objective optimization is not directly applicable in these settings, because it is impossible to optimize multiple conflicting objectives at the same time. Rather, we seek a good tradeoff among the multiple objectives, which can be formalized using the notion of Pareto optimality or Pareto efficiency, a well-known criterion of performance for multi-objective optimization in economics and engineering.

We formulate the multi-objective optimization problem using the general form

$$\min_{x \in X} F(x) = (f_1(x), f_2(x), ..., f_q(x)),$$

where X is a closed subset of $\mathbb{R}^p$, q is the number of objectives, and $F : \mathbb{R}^p \to \mathbb{R}^q$ is a vector valued function. Pareto optimality for the problem with the above form is defined by the following dominance relationship (Abraham, Jain, and Goldberg 2005):

Definition 1 Let x_u, $x_v \in X$ be two decision vectors. $F(x_u)$ *dominates* $F(x_v)$ (denoted by $F(x_u) \prec F(x_v)$) if and only if $f_i(x_u) \leq f_i(x_v)$, for all $i \in \{1,2,...,q\}$ and there exists $j \in \{1,2,...,q\}$ such that $f_j(x_u) < f_j(x_v)$.

Definition 2 A point $x^* \in X$ is said to be *globally Pareto optimal* if and only if there is no $x \in X$ such that $F(x) \prec F(x^*)$. $F(x^*)$ is called *globally efficient* and the image of the set of globally efficient points is called the *Pareto front*.

Multi-objective optimization is especially challenging because the goal is to find a set of solutions that can accurately approximate the Pareto front, rather than search for a single solution. In particular, when the Pareto front is disconnected or is highly nonlinear, we may only be able to find Pareto optimal solutions within a local region. In addition, if the image of X under the mapping F is disconnected, the best we can guarantee with local search methods is the local optimality of the solutions. Local Pareto optimality is formally defined as follows:

Definition 3 A point $x^* \in X$ is said to be *locally Pareto optimal* if and only if there exists an open neighborhood of x^*, $\mathfrak{N}(x^*)$, such that there is no $x \in \mathfrak{N}(x^*) \cap X$ satisfying $F(x) \prec F(x^*)$. $F(x^*)$ is then called *locally efficient* and the image of the set of locally efficient points is called the *local Pareto front*.

The literature contains a number of methods for identifying a set of Pareto optimal solutions in deterministic multi-objective optimization. One of the most widely used approaches to solve multi-objective optimization problems works by converting the multiple objectives into one single objective. The weighted sum method (WS) (Shan and Wang 2005, Cohon 1978) minimizes a convex combination of the multiple objectives. Using different weights, it is possible to find multiple optimal solutions and use them to approximate the Pareto front. However, the points produced by the WS method all lie on the convex part of the Pareto front. Another drawback of the WS method is that different weight combinations often yield the same optimal solution. Audet et al. (2008) focuses particularly on a class of bi-objective problems and applies the mesh adaptive direct search (MADS) method to a series of single objective problems. The algorithm is called BIMADS (Bi-objective MADS) and is designed to generate a set of solutions on both convex and nonconvex parts of the local Pareto front. However, the convergence rate may be slow for high-dimensional problems, a common drawback of direct search methods. Ryu and Kim (2011) proposed a locally convergent trust-region method for the bi-objective black-box optimization problem. This approach was demonstrated numerically to outperform BIMADS on a certain set of problems and produce good performance in both convex and nonconvex settings. One can also apply an evolutionary algorithm combined with the weighted sum method to large-scale, complex problems (Abraham, Jain, and Goldberg 2005, Deb 1999). However, this algorithm has no convergence guarantees, and tends to evaluate a very large number of non-Pareto set points, incurring a high computational cost.

The problem becomes even more complex and challenging with the addition of stochastic noise. The literature on this problem class is very limited, particularly when the decision space of the stochastic multi-objective problem is continuous. We consider black-box problems where we do not have analytic forms for the objective functions $f_i, i = 1, \ldots, q$. The objective values can only be estimated through expensive simulations. The output of these simulations is subject to stochastic variation. We focus on multi-objective stochastic problems with the form

$$\min_{x \in X} \left\{ F(x) = (f_1(x), \ldots, f_q(x)) := E[F(x, \xi)] = E\left[f_1(x, \xi), \ldots, f_q(x, \xi)\right] \right\}. \tag{1}$$

Here, ξ is a random vector defined on a probability space $(\Omega, \mathscr{F}, P)$ with support Ξ. The sample objective functions $f_i(\cdot, \cdot) : \mathbb{R}^p \times \Xi \to \mathbb{R}$, $i = 1, \ldots, q$, are real valued functions and their values are evaluated via simulation. The value of the expected functions are finite and can be estimated via sample averages. We assume that the objective functions $f_1, \ldots, f_q$ are in conflict with each other and have finite minimum values.

We use the sample average approximation (SAA) approach (Shapiro 2003, Shapiro et al. 2009) to approximate the problem (1). Essentially, we replace the values of the expected function by the sample average values, and use deterministic optimization to solve the ensuing problem. The SAA approximation to (1) is given by

$$\min_{x\in X}\left\{\hat{F}_N(x)=\left(\hat{f}_1(x),\ldots,\hat{f}_q(x)\right)=\left(\frac{1}{N}\sum_{l=1}^{N}f_1(x,\xi_l),\ldots,\frac{1}{N}\sum_{l=1}^{N}f_q(x,\xi_l)\right)\right\}, \tag{2}$$

where N is the sample size and $\xi_1,\ldots,\xi_N$ are N independent replications of ξ.

We find non-dominated solutions by modifying the trust-region method for multi-objective deterministic optimization developed by Ryu and Kim (2011), and applying it to the SAA problem (2). In every iteration, we create a trust region and build quadratic approximation functions for the expected objectives, using the sample average values at design points that are selected within the trust region. We then construct a single-objective optimization problem based on these approximations, and solve it to find non-dominated solutions in the trust region. After updating the set of non-dominated solutions, we create a new trust region around the most isolated point, to promote exploration of areas that we have not yet visited.

The present paper builds on Kim and hyun Ryu (2011), the first work to apply the trust-region idea to stochastic multi-objective optimization. However, this work primarily focused on computational issues, and proposed various practical heuristics (such as a variable sampling scheme) to improve the performance of the algorithm. By contrast, this paper focuses on the theoretical aspects of the method, particularly convergence, an issue that was not touched upon by Kim and hyun Ryu (2011). Our main contribution is an outline of a proof of local convergence for SAA applied to multi-objective stochastic approximation. We also present additional numerical results showing the practical potential of the algorithm.

This paper is organized as follows. In Section 2, we provide the convergence results of sample average approximation method for multi-objective problems. Section 3 describes the proposed algorithm for solving stochastic bi-objective optimization problems in detail. In Section 4, we conduct numerical experiments to show the feasibility of our proposed method. Section 5 draws some concluding remarks.

2 CONVERGENCE OF SAA METHOD

Thus far, the SAA method has not been actively studied in the context of multi-objective optimization. We focus on the convergence of the solutions of the SAA problem to those of the true multi-objective problem (MOP). Let $\mathscr{X}^*$ and $\mathscr{X}_N$ be the set of Pareto optimal solutions for the MOP (1) and the SAA problem (2), respectively. The most desirable convergence result that we may want to achieve is the convergence of the approximate solution set $\mathscr{X}_N$ to the true solution set $\mathscr{X}^*$ with increasing sample size N. This result is especially challenging because of the complex structure of the set of Pareto optimal solutions. The sets $\mathscr{X}_N$ and $\mathscr{X}^*$ are often disconnected, with continuous and/or discrete components, particularly when the functions f_i are highly nonlinear with multiple local minima and maxima.

A Pareto-optimal solution is not necessarily a minimizer of any one individual objective function, but is determined by simultaneously comparing all the objective function values with all other solutions. This makes it difficult to determine the Pareto optimality of the limit of $\mathscr{X}_N$. On the other hand, local optimality can be achieved under a simpler condition and has been well studied for a deterministic MOP (Ehrgott et al. 2005). The first-order necessary optimality conditions for MOP are as follows.

Definition 4 Consider MOP (1) and let f_i be continuously differentiable at $\tilde{x}\in X$. If $\tilde{x}$ is locally Pareto optimal, then there exists some $\check{\imath}\in\{1,,\ldots,q\}$ for any d in the tangent cone $T_X(\tilde{x})$ such that $\nabla f_{\check{\imath}}(\tilde{x})^T d\geq 0$. We call the point $\tilde{x}$ a *first-order Pareto solution* for MOP.

We leave the analysis on global convergence for future work. Instead, we focus on the local convergence of the SAA method and develop an algorithm based on a scalarization method for generating the set of optimal solutions. In particular, we use the following form of single-objective optimization proposed by Audet et al. (2008):

$$\min_{x\in X}\phi(x;r)=-\prod_{i=1}^{q}\{(r_i-f_i(x))_+\}^2, \tag{3}$$

where $r = (r_1, \ldots, r_q)$ is a reference point in objective function space $\mathbb{R}^q$. Note that $\phi(x;r)$ is continuously differentiable at $\tilde{x}$ if F is. By changing the reference point r, one can find a series of Pareto optimal solutions in the area of interest. Some efficient points may not be found with this single-objective formulation. However, the numerical results show that this formulation generally works well for various types of Pareto efficient fronts. If all objective functions $f_i, i = 1, \ldots q$ are convex, then a solution of (3) is Pareto optimal. The next proposition gives more general necessary condition of a Pareto optimal solution based on the above formulation (3).

Proposition 1 (Audet et al. (2008)) If there exists a vector $r \in \mathbb{R}^q$ such that $\tilde{x}$ is an optimal solution of the problem (3) with $\phi(\tilde{x};r) < 0$ (or equivalently, $F(\tilde{x}) < r$), then $\tilde{x}$ is Pareto optimal for the MOP. Moreover, assume that $f_i, i = 1, \ldots, q$ are continuously differentiable, and for a given r, $\tilde{x}$ is a first order critical point of $\phi(x;r)$ such that $\phi(\tilde{x};r) < 0$. Then, $\tilde{x}$ is the first order Pareto solution of the MOP.

Let $\xi_1, ..., \xi_N$ be i.i.d. replications of the random vector ξ. Then, for any $x \in X$, the above function $\phi(x;r)$ can be estimated by averaging values of the product of sample functions. The corresponding SAA problem is defined as follows:

$$\min_{x \in X} \hat{\phi}_N(x;r) = \frac{1}{N} \sum_{l=1}^{N} \phi(x, \xi_l; r) \tag{4}$$

where $\phi(x,\xi;r) = -\prod_{i=1}^{q} \{(r_i - f_i(x,\xi))_+\}^2$. Under a set of conditions, we can ensure that the optimal solution to the problem (4) converges to a true Pareto optimal solution. Let $X^*(r)$ and $X_N(r)$ be the set of optimal solutions for the problem (3) and (4), respectively. We define the distance from a point x to a compact set A to be $d(x,A) = \inf_{y \in A} \|x - y\|$. For two compact sets A and B, $\mathbb{D}(A,B) = \sup_{x \in A} d(x,B)$ denotes the deviation of set A from the set B.

Theorem 2 Assume that for a given reference point $r \in \mathbb{R}^q$,

(i) the minimum value of $\phi(x;r)$ is nonzero,
(ii) $X^*(r)$ is nonempty and is contained in a compact set $C \subset X$,
(iii) $f_1, \ldots, f_q$ are continuous and are finite valued on C,
(iv) for N large enough, $X_N(r)$ is nonempty and is contained in C with probability 1 (w.p.1), and
(v) $\hat{\phi}_N(\cdot;r)$ converges to $\phi(\cdot;r)$ w.p.1. as $N \to \infty$ uniformly in $x \in C$.

Then, for N large enough $X_N(r) \subset \mathscr{X}_N$ and $\mathbb{D}(X_N(r), X^*(r)) \to 0$ w.p.1 as $N \to \infty$.

Proof. The proof is immediate from Theorem 5.3 in Shapiro et al. (2009). □

Assumption (v) is called the uniform law of large number, which is the key condition for the convergence result. Assumption (iii) and (v) can be ensured by imposing continuity and integrability conditions to each sample objective functions.

Proposition 3 Suppose that for a given reference point $r \in \mathbb{R}^q$,

(i) for every $z \in \Theta$, $f_i(\cdot, z), i = 1, \ldots, q$ are continuous on a compact set C,
(ii) $f_i(x,\xi), i = 1, \ldots, q$ are dominated by integrable functions in $x \in C$, that is, there exist nonnegative valued integrable functions $g_i(\xi)$ such that for every $x \in C$ $|f_i(x,\xi)| \le g_i(\xi)$ w.p.1, and
(iii) $\mathbb{E}[g_i(\xi)]^{2q} < \infty$.

Then, $f_i(\cdot)$ and $\phi(x;r)$ are finite valued and continuous on C, and $\hat{F}_N(\cdot)$ and $\hat{\phi}_N(\cdot;r)$ converge to $F(\cdot)$ and $\phi(\cdot;r)$, respectively, w.p.1. as $N \to \infty$ uniformly.

Proof. The proof is immediate from Proposition 7 in Shapiro (2003) combined with Hölder's inequality. □

Suppose that we apply a locally convergent algorithm to the SAA problem (4). Then, the best we can guarantee is convergence to the local Pareto optimal solutions for the true MOP. Let S^* and S_N be the set of first-order Pareto solutions of the MOP problem (1) and the SAA problem (2), respectively. The following results ensure the convergence of the first-order Pareto solutions to those of the true problem.

Theorem 4 Suppose that for a given reference point $r \in \mathbb{R}^q$,

(i) X is a compact convex set,
(ii) for every $z \in \Theta$, $f_i(\cdot, z), i = 1, \ldots, q$ are continuously differentiable on a neighborhood of X, and
(iii) the gradient components $\frac{\partial}{\partial x_j} f_i(x, \xi)(j = 1, \ldots, p)$ are dominated by an integrable functions $h_{ij}(\xi)$ with $E[h_{ij}(\xi)]^{2q-1} < \infty$.

Then,

(a) $F(x)$ is continuously differentiable,
(b) $\nabla \hat{F}_N(x) \to \nabla F(x)$ w.p.1. as $N \to \infty$ uniformly, and
(c) $\mathbb{D}(S_N, S^*) \to 0$ w.p.1 as $N \to \infty$.

Proof. To show (a) and (b), first apply Proposition3 to the each component of the gradient and then apply Proposition 3 to the sample gradient function. We prove (c) by contradiction. Suppose that $\mathbb{D}(S_N, S^*) \nrightarrow 0$. Since X is compact, by passing to a subsequence if necessary, we can assume that there exists $x_N \in S_N$ such that for some $\alpha > 0$, $d(x_N, S^*) \geq \alpha$ for all $N \geq 1$, and that x_N tends to a point $x^* \in X$. It follows that $x^* \notin S^*$.

Each x_N satisfies the first order Pareto optimal condition. Since X is convex, for any $u \in X$ there exists some $\check{\imath}$ such that

$$\nabla \hat{f}_{\check{\imath}}(x_N)^T (u - x_N) \geq 0 \text{ w.p.1.} \tag{5}$$

On the other hand, for any $u \in X$ and $i = 1, \ldots, q$,

$$\nabla \hat{f}_i(x_N)^T (u - x_N) \to \nabla f_i(x^*)^T (u - x^*) \tag{6}$$

w.p.1 as $N \to \infty$, by (b). Since $x^* \notin S^*$, for some $u \in X$ and $\varepsilon > 0$, $\nabla f_i(x^*)^T (u - x^*) < -\varepsilon/2$ for all $i = 1, \ldots, q$. By (6), for N large enough, $\|\nabla \hat{f}_i(x_N)^T (u - x_N) - \nabla f_i(x^*)^T (u - x^*)\| < \varepsilon/2$, and hence $\nabla \hat{f}_i(x_N)^T (u - x_N) < 0$ for all $i = 1, \ldots, q$. This is a contradiction to (5).

□

3 BI-OBJECTIVE STOCHASTIC OPTIMIZATION ALGORITHM

In this section, we consider the bi-objective stochastic optimization problem and present an algorithm to generate a set of Pareto optimal solutions based on the SAA single-objective optimization problem (4). Instead of solving (4) multiple times for a pre-determined set of reference point r, we adaptively change r at each iteration to improve the approximate Pareto front generated in previous iterations. The algorithm iteratively applies a trust-region method (Ryu and Kim 2011) to the SAA problem (2) and finds a set of non-dominated points within a local region. At each iteration, the most isolated point is selected among the points that have thus far been determined to be non-dominated. This point is then defined as the current iterate. A trust region centered at the current iterate is determined to maintain the uniformity of the optimal solution set by exploring non-visited areas. Thus, the trust region iteratively moves according to the selected iterate. Several design points in the trust region are chosen using a design of experiment technique, and the sample average values at the design points as well as the iterate are computed. A quadratic regression model for each objective function is constructed based on the sample average values at the design points (Myers, Montgomery, and Anderson-Cook 2009), and a single-objective optimization problem is built to search for non-dominated solutions within the trust region.

3.1 Iterate Determination

We define

$$\mathscr{X}_N^{(k)} = \{x_j^{(k)}, j = 1,2,\ldots,J^{(k)}\}$$

to be the set of non-dominated points returned at the end of iteration $(k-1)$, where $J^{(k)}$ denotes the cardinality of $\mathscr{X}_N^{(k)}$. The non-dominated point $x_j^{(k)}$ is associated with the following four parameters:

- $\hat{F}_j^{(k)} = \left(\hat{f}_1{}_j^{(k)}, \hat{f}_2{}_j^{(k)}\right)$: the vector of sample average objectives evaluated at $x_j^{(k)}$. For $i = 1,2$, $\hat{f}_{ij}^{(k)} = \frac{1}{N}\sum_{l=1}^{N} f_i(x_j^{(k)}, \xi_{il})$.
- $\Delta_j^{(k)}$: Suppose that $x_j^{(k)}$ is detected at some iteration $m < k$. If Δ is the radius of the trust-region computed by Step 6 in Section 3.3 at $x_j^{(k)}$ at the iteration m, $\Delta_j^{(k)} = \Delta$.

While we would like to identify solutions close to the Pareto front, we also want to generate well-spread solutions in order to approximate as much of the Pareto front as possible. To this end, we select the most isolated point $x_c^{(k)}$ in $\mathscr{X}_N^{(k)}$ and search for new solutions around a region centered at the point. We introduce a quantity $\gamma_j^{(k)}$ that indicates the degree of isolation of point $x_j^{(k)}$, based on distance between the objective vectors at $x_j^{(k)}$ and its neighboring points. The distance is estimated with the sample average objective function values at non-dominated points. The procedure to compute the isolation measure $\gamma_j^{(k)}, j = 1,2,\ldots,J^{(k)}$ and to select the most isolated point $x_c^{(k)}$ is as follows:

1. Sort points in $\mathscr{X}_N^{(k)}$ so that $\hat{f}_{11}^{(k)} \leq \hat{f}_{12}^{(k)} \leq \ldots \hat{f}_{1J^{(k)}}^{(k)}$.
2. Let $\delta > 0$ be a user-defined constant. We compute $\gamma_j^{(k)}$ using the below formula:

$$\gamma_j^{(k)} = \begin{cases} 2\|\hat{F}_N(x_1) - \hat{F}_N(x_2)\| & \text{if } i = 1 \\ 2\|\hat{F}_N(x_{J^{(k)}-1}) - \hat{F}_N(x_{J^{(k)}})\| & \text{if } i = J^{(k)} \\ \|\hat{F}_N(x_{i-1}) - \hat{F}_N(x_i)\| + \|\hat{F}_N(x_i) - \hat{F}_N(x_{i+1})\| & \text{otherwise,} \end{cases} \tag{7}$$

   where $\|\cdot\|$ denotes the Euclidean distance.
3. Let $\gamma_{max}^{(k)} = \max\{\gamma_j^{(k)} : j = 1,2,\ldots,J^{(k)}\}$, $j^* = \text{argmax}\{\gamma_j^{(k)} : j = 1,2,\ldots,J^{(k)}\}$, and $\Delta_{tol} > 0$ be the convergence tolerance parameter of the trust-region method.
   - If $\gamma_{max}^{(k)} \geq \delta$, $c = j^*$,
   - else if $\Delta_1^{(k)} > \max\{\Delta_{tol}, \Delta_{J^{(k)}}^{(k)}\}, c = 1$
   - else if $\Delta_{J^{(k)}}^{(k)} > \max\{\Delta_{tol}, \Delta_1^{(k)}\}, c = J^{(k)}$,
   - else $c = j^*$.

If $\gamma_{max}^{(k)} < \delta$, it follows that the solutions in $\mathscr{X}_N^{(k)}$ are all close to each other. In this case, either $x_1^{(k)}$ or $x_{J^{(k)}}^{(k)}$ is selected so that the iterate $x_c^{(k)}$ moves toward the end part of the Pareto front. In this way, the algorithm can generate well-spread solutions while searching for the minimizer of each objective as the iteration k grows.

3.2 Quadratic Regression Model

Once the center point is defined, we would like to find the best trade-off solution between the two conflicting objectives within the neighborhood of the center point. Let

$$\mathbf{B}(x,\Delta) = \{y \in \mathbb{R}^n \ : \ \|y - x\| \leq \Delta\}$$

denote the closed ball centered at x with radius Δ. The trust-region at iteration k is defined by $\mathbf{B}^{(k)} = \mathbf{B}(x_c^{(k)}, \Delta^{(k)})$, where $\Delta^{(k)}$ is the trust region radius determined by Step 3 in Section 3.3.

In order to find the best trade-off solution, we combine the two objectives using the single-objective formulation (4):

$$\hat{f}_3{}^{(k)}(x) = -\frac{1}{N}\sum_{l=1}^{N}\prod_{i=1}^{2}\left\{(r_i^{(k)} - f_i(x,\xi_l))_+\right\}^2$$

where we set $r_i^{(k)} = f_i(x_c^{(k)})$, $i = 1,2$ in the algorithm.

We assume that $\hat{f}_1$ and $\hat{f}_2$ are not easy to optimize, so $\hat{f}_3$ is difficult as well. We would like to construct local surrogate models on the trust region to search for local optimal solutions. The optimization of the surrogate model should be easier than that of the original objective function, and yet the model has to be accurate enough to obtain a reasonably good solution. One commonly used model is a quadratic function. We sample points in $\mathbf{B}^{(k)}$ by using a central composite design (Myers, Montgomery, and Anderson-Cook 2009) and evaluate the sample average objectives at the design point as well as the center point with the sample size N. Let $m_i^{(k)}$ be the quadratic model for $\hat{f}_i, i = 1,2,3$, which can also be written in the following form:

$$\hat{f}_i^{(k)}(x_c^{(k)} + s) \approx m_i^{(k)}(x_c^{(k)} + s) = c_i^{(k)} + s^{\mathrm{T}} g_i^{(k)} + \frac{1}{2} s^{\mathrm{T}} H_i^{(k)} s.$$

where $x_c^{(k)} + s \in \mathbf{B}^{(k)}$, $c_i^{(k)}$ are scalar, and $g_i^{(k)}$ and $H_i^{(k)}$ are the gradient and the Hessian of the quadratic model $m_i^{(k)}$ at $x_c^{(k)}$ $(i = 1,2,3)$.

3.3 Algorithm Description

I. Initialization: Choose the initial point $x^{(0)}$, the initial trust-region radius $\Delta_0 > 0$, and the value of parameters $\Delta_{max} > 0$, $\Delta_{tol} > 0$, $\delta > 0$, $0 \le \eta < 1$, $\tau \in (0,1)$, $\tau_{inc} > 1$, $\mu > \beta > 0$, and $\omega \in (0,1)$. Evaluate $\hat{f}_1(x^{(0)})$ and $\hat{f}_2(x^{(0)})$ and set $\Delta_c^{(0)} = \Delta_0 \in (0, \Delta_{max})$.

Step 1 (next iterate selection) From the previous iteration, the set of non-dominated points $\mathscr{X}_N^{(k)}$ is returned. Select $x_c^{(k)}$ in $\mathscr{X}_N^{(k)}$ according to the procedure described in Section 3.1 and set $\Delta_{icb}^{(k)} = \Delta_c^{(k)}$.

Step 2 (regression models) Using a poised set of sample points, $Y = \{y^1, y^2, \ldots, y^d\}$, in the ball $\mathbf{B}(x_c^{(k)}, \Delta_{icb}^{(k)})$, construct fully linear models $m_i^{icb}, i = 1,2,3$ as described in Section 3.2.

Step 3 (criticality step) If $\mu^{-1}\Delta^{(k)} \le \min_{i=1,2,3} \|g_i^{(k)}\|$, then set $\Delta^{(k)} = \Delta_{icb}^{(k)}$ and $m_i^{(k)} = m_i^{icb}, i = 1,2,3$. Otherwise, proceed as follows.

Initialization : Set $\hat{\imath} = 0$ and $\tilde{m}_i^{(0)} = m_i^{icb}, i = 1,2,3$.

Repeat Set $\hat{\imath} = \hat{\imath} + 1$. As done in Step 2, construct fully linear models $\tilde{m}_i^{(\hat{\imath})}, i = 1,2,3$ in the ball $\mathbf{B}(x_c^{(k)}, \omega^{\hat{\imath}}\Delta_{icb}^{(k)})$:

$$\tilde{m}_i^{(\hat{\imath})}(x_c^{(k)} + s) = \tilde{c}_i^{(\hat{\imath})} + s^{\mathrm{T}}\tilde{g}_i^{(\hat{\imath})} + \frac{1}{2} s^{\mathrm{T}}\tilde{H}_i^{(\hat{\imath})} s.$$

Set $\tilde{\Delta}^{(k)} = \omega^{\hat{\imath}}\Delta_{icb}^{(k)}$.

Until $\tilde{\Delta}^{(k)} \le \min_{i=1,2,3} \mu\|\tilde{g}_i^{(\hat{\imath})}\|$ or $\tilde{\Delta}^{(k)} \le \Delta_{tol}$.

Then set

$$m_i^{(k)} = \tilde{m}_i^{(\hat{\imath})} \text{ and } \Delta^{(k)} = \min\left[\max\left\{\tilde{\Delta}^{(k)}, \min_{i=1,2,3} \beta\|\tilde{g}_i^{(\hat{\imath})}\|\right\}, \Delta_{icb}^{(k)}\right].$$

Step 4 (three single-objective problems) Solve each single objective problem and let

$$x_{i^*}^{(k)} = \arg\min\{m_i^{(k)}(x) : x \in \mathbf{B}(x_c^{(k)}, \Delta^{(k)}) \cap X\},$$

where $i^* = J^{(k)} + d_{tot}^{(k)} + i,\ i = 1,2,3$ and $d_{tot}^{(k)} = d \times \hat{t}$ denotes the total number of evaluated design points in the current iteration.

Step 5 (reduction ratios) Compute $\rho_{i^*},\ i = 1,2,3$, for the model solutions from Step 4 such that

$$\rho_{i*} = \frac{\hat{f}_i(x_c^{(k)}) - \hat{f}_i(x_{i^*}^{(k)})}{m_i(x_c^{(k)}) - m_i(x_{i^*}^{(k)})},\ i = 1,2,3.$$

Compute $\tilde{\rho}_j$ for $j = J^{(k)} + 1, \ldots, J^{(k)} + d_{tot}^{(k)} + 3$, such that

$$\tilde{\rho}_j = \frac{\hat{f}_3(x_c^{(k)}) - \hat{f}_3(x_j^{(k)})}{m_3(x_c^{(k)}) - m_3(x_{3^*}^{(k)})}.$$

Step 6 (trust-region radius update) In the current iteration, the number of newly evaluated points is $(d_{tot}^{(k)} + 3)$. Set a new trust region radius corresponding to design points,

$$\Delta_{J^{(k)}+j}^{(k)} = \|x_c^{(k)} - x_{J^{(k)}+j}^{(k)}\|,\ j = 1, \ldots, d_{tot}^{(k)}.$$

For the three model solutions, $\{x_{1^*}, x_{2^*}, x_{3^*}\}$,

$$\Delta_{i^*}^{(k)} \in \begin{cases} \left[\Delta^{(k)}, \min\{\tau_{inc}\Delta^{(k)}, \Delta_{max}\}\right] & \text{if } \rho_{i^*} \geq \eta \\ \{\tau\Delta^{(k)}\} & \text{otherwise.} \end{cases}$$

Step 7 (non-dominated solution set update) Compare the vector values $\hat{F}(x)$, $x \in \{x_j^{(k)}, j = 1,2, \ldots, (J^{(k)} + d_{tot}^{(k)} + 3)\}$ and determine the set of non-dominated points $\mathscr{X}_N^{(k+1)}$.

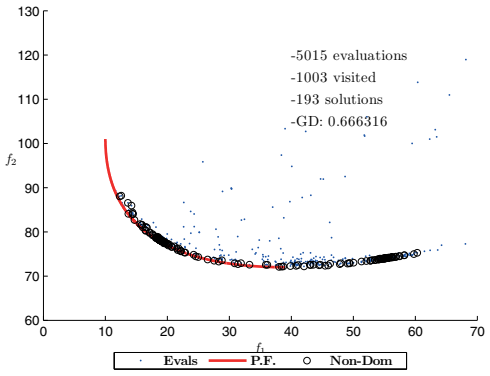
Step 8 (iteration evaluation) If $\max\{\tilde{\rho}_j : j = J^{(k)} + 1, \ldots, (J^{(k)} + d_{tot}^{(k)} + 3)\} < \eta$ and $\tau\Delta^{(k)} > \Delta_{tol}$, set $x_c^{(k+1)} = x_c^{(k)}$, $\Delta_{icb}^{(k+1)} = \tau\Delta^{(k)}$, increase k by one, and then go to Step 2. Otherwise, increase k by one and go to Step 1.

The deterministic version of the above algorithm is guaranteed to converge under a set of regularity conditions such as the Lipschitz continuity of objective functions and full linearity of the surrogate models m_i (see Ryu and Kim (2011)). If for fixed $z \in \Xi$, the sample objective functions $f_i(\cdot, z)(i = 1,2)$ are Lipschitz continuous with Lipschitz coefficient $K_i(z)$ and $K_i(\xi)$ is integrable, the objective functions f_i are also Lipschitz continuous.

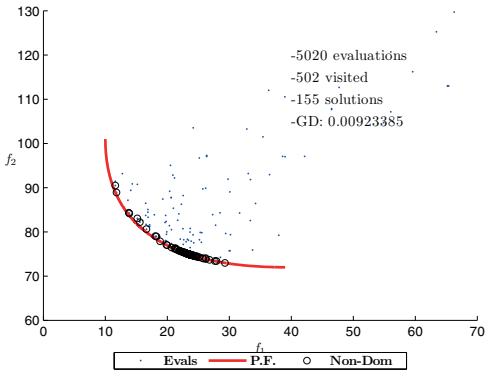
4 NUMERICAL EXPERIMENTS

In this section, we test the proposed algorithm with several sampling schemes. The test problem, taken from (Wilson, Cappelleri, Simpson, and Frecker 2001), is an unconstrained deterministic bi-objective problem with a convex Pareto front. After adding noise to the decision variable x, our problem is formulated as follows:

$$\begin{aligned} \text{Minimize:} \quad & E[f_1(x_{(1)}, x_{(2)}, \xi)] = E(x_{(1)} - 2\xi_{(1)})^2 + (x_{(2)} - \xi_{(2)})^2] \\ & E[f_2(x_{(1)}, x_{(2)}, \xi)] = E[(x_{(1)})^2 + (x_{(2)} - 6\xi_{(3)})^2], \end{aligned}$$

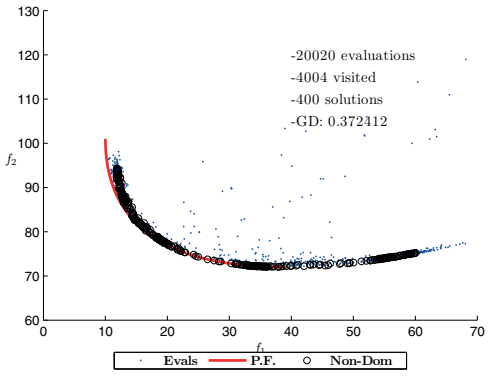


(a) $N = 5$

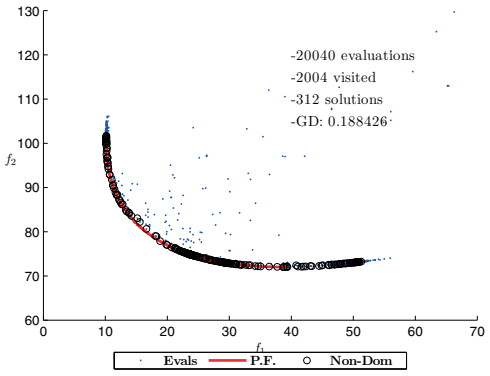


(b) $N = 10$

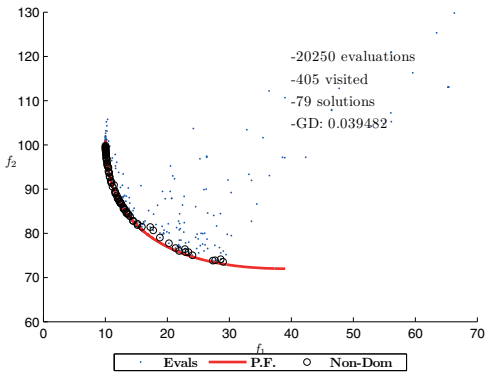
Figure 1: Performance comparisons with 5,000 function evaluations.



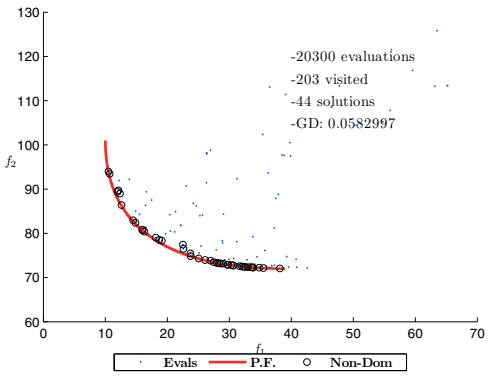
(a) $N = 5$



(b) $N = 10$



(c) $N = 50$



(d) $N = 100$

Figure 2: Performance comparisons with 20,000 function evaluations.

where $\xi = [\xi_{(1)}, \xi_{(2)}, \xi_{(3)}]$, and $\xi_{(1)}, \xi_{(2)}, \xi_{(3)}$ are independent and identically distributed chi-square random variables with degree 1. The starting point is $x^{(0)} = [5,5]$ and the initial parameters are set by $\Delta^{(0)} = 0.8$, $\Delta_{tol} = 0.001$, $\alpha = 0.98$, $\eta = 0.5$, $\tau = 0.7$, $\tau_{inc} = 1$, and $\delta = 0.001$. The Pareto front is approximated by evaluating 401×701 uniformly-spaced points on $[-2,2] \times [0,7]$. We find a set of solutions $\mathscr{H}$ of around $2,500$ non-dominated solutions that are uniformly-spaced.

To evaluate our method, we use the generational distance (GD) criterion (Veldhuizen and Lamont 1998). Suppose that the solution set is $\mathscr{H} = \{x_1, \ldots, x_e\}$. Then, the GD is computed by

$$GD = \frac{\sqrt{\sum_{j=1}^{e} \left\{ \min_{x_j^* \in \mathscr{H}} \|F(x_j) - F(x_j^*)\| \right\}^2}}{e}.$$

This is a measure of the average distance between the objective value at the obtained solution and the true Pareto front. Hence, smaller GD is preferable. Note that GD$= 0$ indicates that all the generated solutions are Pareto optimal.

We tested the algorithm in Section 3.3 with sample size $N = 5, 10, 50$ and 100. The computation time is measured in terms of function evaluations, and Figure 1 and 2 present the performance comparisons with $5,000$ and $20,000$ function evaluations, respectively. All dots present the true objective function values at all the visited points. Small and large dots represent the points determined to be dominated and non-dominated based on the sample average function values, respectively. The solid line is the true Pareto front. When $N = 5$ and 10, quite a number of points are generated near the Pareto front within $5,000$ function evaluations (Figure 1, (a) and (b)). With $20,000$ function evaluations, however, a large number of dominated points cannot be screened out due to the sampling error, and they are eventually wrongly determined as non-dominated points (Figure 2, (a) and (b)). In particular, with $N = 5$, nearly 50% of solutions generated from the algorithm are dominated points. On the other hand, with $N = 50$ and 100, almost all the generated points are near the Pareto front (Figure 2, (c) and (d)). However, the algorithm was not able to generate solutions that cover all range of Pareto front. When the sample size N is large, the algorithm evaluates a smaller number of points, and thus cannot fully explore the entire Pareto front.

We tested 100 replications of the SAA problem with $N = 5, 10, 50$ and 100 for the average performance comparisons. Table 1 present the mean and standard error of GD and Table 2 present the number simulation runs that returned a set of solutions with a certain level of GD. The results show that he efficiency of the algorithm significantly depends on the sample size. From the both tables, we can observe that with $5,000$ function evaluations, $N = 10$ performs the best, and with $20,000$ function evaluations, $N = 50$ performs significantly better than others. This implies that the sample size should be carefully determined taking into account the computational budget.

Table 1: GD performance comparisons.

N_k	5,000 evaluations		20,000 evaluations	
	mean of GD	standard error	mean of GD	standard error
5	0.4998	0.9987	0.3177	0.5762
10	0.2920	0.4895	0.2182	0.3003
50	0.5108	0.4661	0.1532	0.3395
100	9.6350	2.2765	0.1743	0.2965

5 CONCLUSION

We developed the framework of the SAA method for MOP and showed the convergence of the SAA method under a set of fairly general regularity conditions. We applied an iterative algorithm for bi-objective

Table 2: The number of runs with GD less than 0.1, 0.5, and 1, with 100 independent runs.

N_k	5,000 evaluations			20,000 evaluations		
	GD<0.1	GD <0.5	GD <1	GD <0.1	GD <0.5	GD<1
5	23	81	90	47	83	90
10	47	86	91	50	86	97
50	6	67	88	63	94	98
100	0	0	0	56	92	98

stochastic optimization problems, based on the trust region method, to the SAA problems. The algorithm does not require any strong modeling assumptions, and has great potential to work well in various real-world settings. The numerical results show that the our proposed method is feasible, and can perform robustly with a large enough size N.

A subject for future work is the convergence of global Pareto solutions in a general context. To improve the finite time performance of the algorithm, the sample size should be carefully determined with consideration for the trade-off between sampling and optimization errors. The difference between the solutions obtained from SAA and the solutions to the true problems can be reduced by taking a larger sample size. On the other hand, as the number of iterations grows, the distance between solutions from each iteration and the Pareto front decreases. We can consider an algorithm to solve a sequence of SAA problems with increasing sample size. In the early stages of the algorithm, given a fixed number of function evaluations, we use a smaller sample size to take a large number of iterations, allowing us to evaluate the objective functions at more points with lower accuracy. In the later stages, with a larger sample size, we make a greater effort to reduce the sample variance and find more accurate solutions. Currently, we are analyzing these two errors for a certain class of problems and developing a heuristic sampling scheme that may work robustly for those problems.

ACKNOWLEDGMENTS

This research is supported by Singapore MOE Academic Research Fund R-266-000-049-133.

REFERENCES

Abraham, A., L. C. Jain, and R. Goldberg. 2005. *Evolutionary multiobjective optimization: theoretical advances and applications*. London: Springer-Verlag.

Audet, C., G. Savard, and W. Zghal. 2008. "Multiobjective Optimization Through a Series of Single-Objective Formulations". *SIAM Journal on Optimization* 19:188–210.

Cohon, J. L. 1978. *Multiobjective Programming and Planning*. New York: Academic Press.

Deb, K. 1999. "Multi-objective genetic algorithms: Problem difficulties and construction of test problems". *Evolutionary Computation* 7:205–230.

Ehrgott, M., J. Figueira, and S. Greco. 2005. *Multiple Criteria Decision Analysis: State of the Art Surveys*. Boston, Dordrecht, London: Springer Verlag.

Kim, S., and J. hyun Ryu. 2011. "A Trust-Region Algorithm for Bi-Objective Stochastic Optimization". In *Proceedings of the 2011 International Conference on Computational Science*, Volume 4, 1422 – 1430: Elsevier.

Markowitz, H. 1991. *Portfolio selection: efficient diversification of investments*. Malden, Massachusetts: Blackwell.

Myers, R., D. Montgomery, and C. Anderson-Cook. 2009. *Response surface methodology: process and product optimization using designed experiments*. Wiley series in probability and statistics. Wiley.

Ryu, J., and S. Kim. 2011. "A Trust Region Method Based on Derivative-Free Models for Multiobjective Optimization Problems". *Submitted for publication*.

Shan, S., and G. G. Wang. 2005. "An Efficient Pareto Set Identification Approach for Multiobjective Optimization on Black-Box Functions". *Journal of Mechanical Design* 127 (5): 866–874.

Shapiro, A. 2003. "Monte Carlo Sampling Methods". In *Stochastic Programming*, edited by A. Ruszczynski and A. Shapiro, Volume 10 of *Handbooks in Operations Research and Management Science*, 353 – 425. Elsevier.

Shapiro, A., D. Dentcheva, and A. Ruszczynski. 2009. *Lectures on Stochastic Programming: Modeling and Theory*. MPS-SIAM Series on Optimization.

Tappeta, R., J. Renaud, and J. Rodríguez. 2002. "An interactive multiobjective optimization design strategy for decision based multidisciplinary design". *Engineering Optimization* 34 (5): 523–544.

Veldhuizen, D. V., and G. Lamont. 1998. "Multiobjective evolutionary algorithm research: A history and analysis". Technical Report TR-98-03, Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Writhe-Patterson AFB, OH.

Wilson, B., D. Cappelleri, T. Simpson, and M. Frecker. 2001. "Efficient Pareto frontier exploration using surrogate approximations". *Optimization and Engineering* 2:31–50.

AUTHOR BIOGRAPHIES

SUJIN KIM is an Assistant Professor in the Department of Industrial and Systems Engineering at the National University of Singapore. She received her Ph.D. degree in Operations Research from Cornell University in 2006. Her research concerns simulation methodology and stochastic simulation-based optimization, with applications in electric power and health service systems. Her e-mail address is iseks@nus.edu.sg.

JONG-HYUN RYU is a Research Fellow in the Department of Industrial and Systems Engineering at the National University of Singapore. He received his Ph.D. degree in Industrial Engineering from Purdue University in 2010. His research interests include simulation optimization, stochastic process and modeling, and quality control. His email address for these proceedings is iserj@nus.edu.sg.