Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

MULTIPLE OBJECTIVE PROBABILISTIC BRANCH AND BOUND FOR PARETO OPTIMAL APPROXIMATION

Hao Huang

Industrial and Systems Engineering
University of Washington
Seattle, WA 98195, USA

Zelda B. Zabinsky

Industrial and Systems Engineering
University of Washington
Seattle, WA 98195, USA

ABSTRACT

We present a multiple objective simulation optimization algorithm called multiple objective probabilistic branch and bound (MOPBnB) with the goal of approximating the efficient frontier and the associated Pareto optimal set in the solution space. MOPBnB is developed for both deterministic and noisy problems with mixed continuous and discrete variables. When the algorithm terminates, it provides a set of non-dominated solutions that approximates the Pareto optimal set and the associated objective function estimates that approximate the efficient frontier. The quality of the solutions is statistically analyzed using a measure of distance between solutions to the true efficient frontier. We also present numerical experiments with benchmark functions to visualize the algorithm and its performance.

1 INTRODUCTION

Decision makers are commonly faced with multiple objectives when evaluating real world systems (Jones, Mirrazavi, and Tamiz 2002). Instead of aggregating the multiple objectives to a single objective with weights, decision makers want to understand the trade-offs between different objectives. The Pareto optimal set provides a family of solutions with objective function values that are not dominated by other solutions. The set of Pareto optimal solutions provides important information that allows the decision makers to determine the final decision. There are many multiple objective optimization studies (Bianchi et al. 2008), especially using evolutionary algorithms such as the genetic algorithm (Deb 2001, Jones, Mirrazavi, and Tamiz 2002), but only limited studies used these algorithms to develop interactive simulation optimization algorithms (Tekin and Sabuncuoglu 2004). Some of the multiple objective simulation optimization algorithms were meta-heuristic based (Lin, Sir, and Pasupathy 2013). Meta-heuristic based multiple objective algorithms could be applied to large scale problems and different types of variables depending on the meta-heuristic algorithm. However, meta-heuristic based multiple objective algorithms were mostly numerically tested without statistical analysis. For our proposed algorithm, we derive probability bounds on the quality of solutions. Our algorithm is designed to provide a set of solutions that is defined by hyper-rectangles in the solution space, in order to provide more insights than obtaining only a few sampled solutions.

Another group of studies, categorized as ranking and selection, can efficiently identify the non-dominated solutions in a finite set (Lee et al. 2004, Lee et al. 2008, Lee et al. 2010). The multiple objective ranking and selection algorithms provided rigorous statistical quality of the solutions by the probability of correct selection. Ranking and selection is an effective approach when the number of alternatives is relatively small. However, it is not efficient when there are continuous variables or a very large number of alternatives. Therefore, our proposed algorithm focuses on mixed-integer/continuous problems.

The probabilistic branch and bound (PBnB) algorithm was first presented in the Winter Simulation Conference 2011 (Zabinsky et al. 2011), and extended to approximate a desired level-set for single objective problems (Huang and Zabinsky 2013). The algorithm presented here extends PBnB by considering multiple

objectives and approximating the set of Pareto optimal solutions. Hence, we refer to the new algorithm as multiple objective probabilistic branch and bound (MOPBnB).

A simulation often has several performance measures that are output from a single run. Our algorithm considers a black box simulation that provides m measures as multiple objectives. When we specify r replications, we are intending that the simulation will be repeated r times, and the simulation provides m measures for each replication.

In Section 2, we describe the details of MOPBnB. In Section 3, we derive probability bounds for the performance of the algorithm. Several numerical results are presented in Section 4, and conclusions in Section 5.

2 Multiple Objective Probabilistic Branch and Bound

The goal of the algorithm is to find an approximation to the efficient frontier in the objective space and the associated Pareto optimal set in the solution space. The multiple objective optimization problem is in the following form:

$$(\mathscr{P}) \ \min_{x\in S}\{f_1(x),\ldots,f_m(x)\}, \tag{1}$$

where the objective functions can only be estimated by $f_l(x) = E_\Xi[g_l(x,\xi_x)], l = 1,\ldots,m$ and $g_l(x,\xi_x)$ is a noisy function and ξ_x is the noise term. We assume the distribution of the noise term ξ_x may be different for each x, but is normally distributed. The feasible set S is assumed to be a hyper-rectangle in n dimensions, to simplify the partition scheme. The variables may be real-valued or integer valued in the hyper-rectangle, with upper and lower bounds, $l_i \le x_i \le u_i$ for a continuous variable, and $x_i \in \{l_i, l_i+1,\ldots,u_i\}$ for an integer variable.

In a single objective problem, we can define the quantile of the objective, as in Zabinsky et al. (2011) and Huang and Zabinsky (2013), to represent the desirable performance. However, with multiple objectives, the definition of the desirable performance is more difficult to characterize. Therefore, we define an indicator $D(x)$, the minimum distance to the true efficient frontier in the objective space, for every x as follows,

$$D(x) = \inf\{||f(x) - f(x^*)|| \ | x^* \in S_e, f(x^*) \text{ dominates } f(x)\}, \tag{2}$$

where S_e is the Pareto set and $||f(x) - f(x^*)||$ is the Euclidean distance in the objective space. Other distance measures besides Euclidean distance may be used in the definition of $D(x)$ and the theoretical analysis still holds. Recognizing that $D(x)$ is not able to be evaluated, we only use it in the analysis to define the desirable performance.

We use $D(x)$ to define a target set $L(\delta,S)$, which can be interpreted as a relaxation of the true Pareto optimal set, for $0 < \delta < 1$. The target set $L(\delta,S)$ is defined in terms of the associated quantile $y(\delta,S)$. We also use $y(p,\sigma)$ and $L(p,\sigma)$ in the analysis of the algorithm for $0 < p < 1$ and $\sigma \subseteq S$. Let $y(p,\sigma)$ be the p quantile of value $D(x)$ where $x \in \sigma$, defined as

$$y(p,\sigma) = \underset{y\in\{D(x):x\in\sigma\}}{\arg\min} \ \{P(D(X) \le y | X \in \sigma) \ge p\}, \text{ for } 0 < p < 1, \tag{3}$$

where X is uniformly distributed in σ. Furthermore, let $L(p,\sigma)$ be a set of best p solutions, defined as

$$L(p,\sigma) = \{x \in \sigma : D(x) \le y(p,\sigma)\}, \text{ for } 0 < p < 1. \tag{4}$$

MOPBnB provides two types of results. First, the set of non-dominated points that have been sampled from S by the algorithm and their associated estimated objective function values are provided and stored in NS_k at iteration k. Second, a collection of hyper-rectangles that approximates the Pareto optimal set is provided, and called the retained region, denoted Σ_k, at iteration k. In Section 3, we derive probability bounds to represent the quality of the solutions. Next, we present the MOPBnB algorithm, incorporating the sample size from Zabinsky et al. (2011) and the ordering method from Huang and Zabinsky (2013).

Multiple Objective Probabilistic Branch and Bound (MOPBnB)

Step 0. Initialize: Set user-defined parameters $\delta, \alpha, R^o, \varepsilon$, and $B \geq 2$. Partition S into B subregions, $\sigma_1, \ldots, \sigma_B$, and set $\Sigma_1 = \{\sigma_1, \ldots, \sigma_B\}$, $\widetilde{\Sigma}_1 = S$, $\alpha_1 = \frac{\alpha}{B}$, $NS_k = \phi$, $R_0 = R^o$, and $k = 1$.

Step 1. Sample: For each subregion $\sigma_i \in \Sigma_k$, $i = 1, \ldots, ||\Sigma_k||$, uniformly sample additional points such that the total number of points in σ_i is

$$N_k = \left\lceil \frac{\ln \alpha_k}{\ln(1-\delta)} \right\rceil. \tag{5}$$

Denote the sample points in σ_i by $x_{i,j}$, for $j = 1, \ldots, N_k$ and $i = 1, \ldots, ||\Sigma_k||$. For each $x_{i,j} \in \sigma_i$, $j = 1, \ldots, N_k$ and $i = 1, \ldots, ||\Sigma_k||$, perform R_{k-1} replications to obtain $g_l(x, \xi_x^r)$ for $l = 1, \ldots, m$, $r = 1, \ldots, R_{k-1}$, and evaluate the sample mean and sample variance for each objective l,

$$\hat{f}_l(x_{i,j}) = \frac{\sum_{r=1}^{R_{k-1}} g_l(x_{i,j}, \xi_x^r)}{R_{k-1}} \text{ and } S^2_{\hat{f}_l}(x_{i,j}) = \frac{1}{(R_{k-1}-1)} \sum_{r=1}^{R_{k-1}} (g_l(x_{i,j}, \xi_x^r) - \hat{f}_n(x_{i,j}))^2 \text{for } l = 1, \ldots, m. \tag{6}$$

Step 2. Calculate additional replications: For $j = 1, \ldots, N_k$ and $i = 1, \ldots, ||\Sigma_k||$ order each objective function l for $l = 1, \ldots, m$ so that

$$\hat{f}_l(z_{(1)}) \leq \hat{f}_l(z_{(2)}) \leq \cdots \leq \hat{f}_l(z_{(||\Sigma_k|| N_k)}), \text{ where } z_{(q)} = \underset{x_{i,j} \in \widetilde{\Sigma}_k \setminus \bigcup_{l=1}^{q-1} z_{(l)}}{\arg\min} \hat{f}_l(x_{i,j}).$$

Let $d_{i,l} = \hat{f}_l(z_{(i+1)}) - \hat{f}_l(z_{(i)})$, where $i = 1, \ldots, N_k - 1$. Determine $d^* = \min_{i=1,\ldots,N_k-1, l=1,\ldots,m} d_{i,l}$ and $S^{*2} = \max_{i=1,\ldots,N_k, l=1,\ldots,m} S^2_{\hat{f}_l}(z_{(i)})$. Calculate the updated replication number $R_k = \max\left\{R_{k-1}, \left(\frac{\tilde{z}(\alpha_k/2) S^*}{d^*/2}\right)^2\right\}$, where $\tilde{z}(\alpha_k/2)$ is the $1 - \alpha_k/2$ quantile of the standard normal distribution. Perform $R_k - R_{k-1}$ more replications for each sample point. Re-estimate the performance of objective l of each sample point with R_k replications by $\hat{f}_l(x_{i,j}) = \frac{\sum_{r=1}^{R_k} g_l(x_{i,j}, \xi^r_{x_{i,j}})}{R_k}$.

Step 3. Identify current non-dominated set: Implement an existing comparison algorithm (Kung, Luccio, and Preparata 1975) to find the non-dominated sample points by $\hat{f}_l(x_{i,j})$ and update NS_k with all non-dominated points and their estimated function values $\hat{f}_l$.

Step 4. Update: For each i, $i = 1, \ldots, ||\Sigma_k||$, update the pruning indicator functions P_i,

$$P_i = \begin{cases} 1, \text{ if } \sigma_i \cap NS_k = \phi \\ 0, \text{ otherwise.} \end{cases} \tag{7}$$

Also, set $\alpha_{k+1} = \frac{\alpha_k}{B}$, $\varepsilon_{k+1} = \frac{\varepsilon_k}{B}$. If $P_i = 0$, and if σ_i is branchable, then partition σ_i to $\bar{\sigma}_i^1, \ldots, \bar{\sigma}_i^B$ and update the current set of subregions

$$\Sigma_{k+1} = \{\bar{\sigma}_i^j : P_i = 0, j = 1, \ldots, B\} \text{ and } \widetilde{\Sigma}_{k+1} = \bigcup_{i: P_i = 0} \left(\bigcup_{j=1}^{B} \bar{\sigma}_i^j \right).$$

A subregion is unbranchable when the longest Euclidean distance of a subregion is less than ε.

Step 5. Stopping Condition: If all subregions $\sigma_i \in \Sigma_k$ are not branchable, terminate the algorithm. Otherwise, $k = k + 1$ and return to **Step 1.**

Users need to determine several parameters: δ, $0 < \delta < 1$, the desired level of quantile; α, $0 < \delta < 1$ to determine the probability bounds involving $1 - \alpha$ on the quality of solutions; R^o as a minimal number of

replications, $R^o \geq 1$; $\varepsilon > 0$, to determine the longest Euclidian distance of a subregion that is considered unbranchable, and B, $B \geq 2$ to determine the partition scheme.

In Step 1 and Step 2, MOPBnB samples uniformly in the current subregions and orders the samples with their estimated function values. A two-stage procedure is applied that is based on a method from Bechhofer, Dunnett, and Sobel (1954) and Gibbons, Olkin, and Sobel (1977) to rank the samples correctly under noise. The two-stage procedure to determine the replications is extended from Huang and Zabinsky (2013) for multiple objectives. The first-stage replications R_{k-1} are evaluated in Step 1. The original ranking procedure assumes that there is a common variance σ^2 but we do not. Step 2 uses the first-stage replications to estimate these differences, d_i for $i = 1, \ldots, N_k - 1$. We use the maximum sample variance to update the replication number for the second-stage. Consequently, additional replications may be necessary.

In Step 3, the algorithm applies any existing method to compare the m objective function estimates for all samples in the retained subregions to eliminate dominated solutions. In this paper, we use the widely used Kung's algorithm (Kung, Luccio, and Preparata 1975). The method will find the non-dominated samples from the retained subregions.

In Step 4, the algorithm updates each subregion to be pruned or further branched into smaller subregions. For each subregion σ_i, the dimension with longest length is chosen for partitioning. For a continuous variable, the length is defined as the difference of the bounds. For a discrete variable, the length is defined as the number of elements. For a continuous variable, partition the associated dimension into even sized B subregions. For a dimension associated with a discrete variable, if it is not possible to partition evenly, then partition into unequal sized subregions ordered by size. For example, suppose the hyper-rectangle for σ_i in two dimensions has a continuous variable x_1, $x_1 \in [1,10]$, and an integer variable x_2, $x_2 \in \{1,2,3,4,5\}$. The length of σ_i on dimension 1 is nine, and on dimension 2 is five, so dimension 1 will be partitioned. And for $B = 2$, the resulting partition on dimension 1 is $[1,5.5]$ and $[5.5,10]$. If dimension 2 is partitioned with $B = 2$, the result is $\{1,2\}$ and $\{3,4,5\}$.

MOPnB follows the same stopping condition as Zabinsky et al. (2011) and Huang and Zabinsky (2013). This stopping condition assures the algorithm will terminate in a finite number of iterations, however users may wish to terminate earlier.

3 Performance Analysis

In this section, we analyze the performance of MOPBnB by deriving probability bounds to indicate the quality of solutions. First, we consider the deterministic case without the noise term. Theorem 1 is analogous to Lemma 1 in Zabinsky et al. (2011) where the single objective function is replaced with the distance function $D(x)$, representing the closeness of the estimated function values of x to the true efficient frontier. The extension is allowed because our use of order statistics has minimal assumptions on the function. In Theorem 2, we derive the quality of the retained region Σ_k at iteration k with no noise. The impact of the noisy function is discussed in Theorems 3 and 4. These are used in Corollary 5 and Theorem 6.

The main results providing probability bounds on the quality of solutions are stated in Corollary 5 and Theorem 6. Corollary 5 is the noisy version of Theorem 2 and gives probability bounds that we correctly retain subregions in our approximation of the Pareto optimal set because they intersect the true Pareto optimal set. Theorem 6 shows that the concentration of points with objective function values close to the true efficient frontier increases with iteration, with probability bounds.

Theorem 1 (c.f. Zabinsky et al. (2011)) Let $y(\delta, \sigma_m)$ be the δ quantile as defined in (3) for some subregion $\sigma_m \in \Sigma_k$ and $0 < \delta < 1$. Suppose $N^* = \left\lceil \frac{\ln \alpha_k}{\ln(1-\delta)} \right\rceil$ and $X_{m,1}, \ldots, X_{m,N^*}$ are sampled i.i.d. uniformly from σ_m. Let $D(X_{m,(1)})$ be the minimum of $D(X_{m,1}), \ldots, D(X_{m,N^*})$, then

$$P(D(X_{m,(1)}) < y(\delta, \sigma_m)) \geq 1 - \alpha. \tag{8}$$

Using (8), we derive the quality of the approximate Pareto optimal set Σ_k in Theorem 2.

Theorem 2 Suppose MOPBnB has progressed to the current k^{th} iteration on Problem $(\mathscr{P})$ where there is no noise, with $\alpha_1 < \alpha$. At iteration k of MOPBnB, let the number of sample points be given as in Step 1,

$$N_k = \left\lceil \frac{\ln \alpha_k}{\ln(1-\delta)} \right\rceil, \tag{9}$$

and let σ_p^k denote the pruned region at iteration k, then

$$P\left(\left(S \setminus \bigcup_{h=1}^{k} \sigma_p^h \right) \bigcap L(\delta, S) \neq \phi \right) \geq 1 - \alpha. \tag{10}$$

Proof. Initially, PBnB branches S into M subregions $\sigma_1, \ldots, \sigma_M$ and samples $N_1 = \left\lceil \frac{\ln \alpha_1}{\ln(1-\delta)} \right\rceil$ on each subregion. Therefore, the total number of sample points from all subregions at first iteration is $M \left\lceil \frac{\ln \alpha_1}{\ln(1-\delta)} \right\rceil$. Within the $M \left\lceil \frac{\ln \alpha_1}{\ln(1-\delta)} \right\rceil$ samples, there is a sample, $X_{S,(1)}$, which performs best with regard to our distance metric, $D(x)$. Note

$$M \left\lceil \frac{\ln \alpha_1}{\ln(1-\delta)} \right\rceil \geq M \left\lceil \frac{\ln \alpha}{\ln(1-\delta)} \right\rceil \geq \left\lceil \frac{\ln \alpha}{\ln(1-\delta)} \right\rceil, \tag{11}$$

thus we know

$$P(D(X_{S,(1)}) < y(\delta, S)) \geq 1 - \alpha \tag{12}$$

by applying Theorem 1 to S. This indicates that $L(\delta, S)$ is sampled by at least one of the sample points in the first iteration with $1-\alpha$ probability. Furthermore, later iterations could have sampled points that dominate the best point of the first iteration, but the non-dominated points will also be in $L(\delta, S)$ with at least $1-\alpha$ probability. Also, non-dominated points will not be pruned by the algorithm. Hence, the intersection of the retaining subregions and $L(\delta, S)$ is non-empty with probability at least $1-\alpha$, and the proof is completed. □

The previous analysis assumes the objective functions $f_l(x)$ are not noisy. In Theorem 3, we consider the objectives that can only be estimated by $\hat{f}_l(x)$. Therefore, in order to achieve the quality in Theorem 2, we seek a correct ordering of the different objective function values to eliminate the dominated solutions, and subregions. Theorem 3 provides rigorous probability bounds for correctly ordering at iteration k, where Theorem 4 combines all iterations from 1 to k and gives a probability bound of $1-\alpha$ on the correct ordering.

We use the analysis of a two-stage replication approach, by Bechhofer, Dunnett, and Sobel (1954), in Theorem 3, which has following assumption:

(A1) The noisy function is normally distributed with an unknown common variance σ^2, and at each solution $z_j \in S$, the variance can be expressed as $V(z_j) = a_j \sigma^2$ where a_j is a known constant for each j.

In MOPBnB, the constants a_j are not known, hence, we implement a modified two-stage replication approach (described after Theorem 3, c.f. Huang and Zabinsky (2013)).

Theorem 3 (cf. Bechhofer, Dunnett, and Sobel (1954)) With Assumption (A1), the probability of correct ordering all samples in the current region at iteration k is

$$P(\hat{f}_l(Z_{(1)}^k) \leq \hat{f}_l(Z_{(2)}^k) \leq \cdots \leq \hat{f}_l(Z_{(N_k)}^k) | f_l(Z_{(1)}^k) \leq f_l(Z_{(2)}^k) \leq \cdots \leq f_l(Z_{(N_k)}^k)) \geq 1 - \alpha_k, \tag{13}$$

given that we have $a_j R_0$ as the first stage replication number for each sample point to estimate the common variance by $S_0^2 = \frac{1}{N_k} \sum_{i=1}^{N_k} \frac{1}{a_j} S_{\hat{f}}^2(z_{(j)})$ and set up the minimum difference desired to be separated as d^*, and sampling $R_k = \max\{a_j R_0, 2(\frac{h S_0^*}{d^*/2})^2\}$ in the second stage of the procedure, where h is a value in the H c.d.f. of a multivariate student t's distribution with $N_k - 1$ dimensions such that $H(h) = 1 - \alpha_k$.

The algorithm intends to find the non-dominated points by correctly ordering points by their objective function values. Although the objective function values could be correlated, the comparison of the function values is separated for each objective function. Therefore, the probability of ordering m objectives correctly for a current iterations is greater than the probability of correct ordering of each objective separately, bounded by $(1-\alpha_k)^m \geq 1-m\alpha_k$. For any iteration k, we take a conservative approach (c.f. Huang and Zabinsky (2013)) to achieve the $1-m\alpha_k$ probability of correct ordering by separating each estimated performance mean $\hat{f}(z_{(i)})$ with its neighbor by the smallest difference $d^* = \min_{i=1,\ldots,N_k-1,l=1,\ldots,n} d_{i,l}$ of any two neighbors. We also use the largest variance $S^{*2} = \max_{i=1,\ldots,N_k,l=1,\ldots,n} S^2_{\hat{f}_l}(z_{(i)})$ so that all ordering is conservative. The implemented two-stage procedure for any iteration k is

1. Implement R_{k-1} replications to estimate the function value $\hat{f}(x)$ and sample variance $S^2_{\hat{f}}(x)$ and calculate the differences $d_{i,l}$ between ordered samples $\hat{f}_l(z_{(i)})$ and $\hat{f}_l(z_{(i+1)})$ for $i = 1,\ldots,N_k-1$ and $l = 1,\ldots,n$.
2. Implement additional replications $R_k - R_{k-1}$ where $R_k = \max\{R_{k-1}, (\frac{\tilde{z}(\alpha_k/2)S^*}{d^*/2})^2\}$.

The following Theorem 4 considers the assumptions and rigorous situation in Theorem 3 to derive a rigorous bound for iteration 1 to k.

Theorem 4 With Assumption (A1), the probability of correct ordering from iteration 1 to iteration k is

$$P\left(\bigcap_{q=1,\ldots,k,l=1,\ldots,m}\left(\hat{f}_l(Z^q_{(1)}) \leq \hat{f}_l(Z^q_{(2)}) \leq \cdots \leq \hat{f}_l(Z^q_{(N_q)}) | f_l(Z^l_{(1)}) \leq f_l(Z^q_{(2)}) \leq \cdots \leq f_l(Z^q_{(N_q)})\right)\right) \geq 1-m\alpha, \tag{14}$$

where $f_l(Z^q_{(j)})$ is the j^{th} ordered sampled point at iteration q by objective l.

Proof. The probability of ordering m objective functions correctly through iteration k is bounded by the individual probability of correct ordering for each iteration and each objective function, given by

$$\begin{aligned} &P\left(\bigcap_{q=1,\ldots,k,l=1,\ldots,m}\left(\hat{f}_l(Z^q_{(1)}) \leq \hat{f}_l(Z^q_{(2)}) \leq \cdots \leq \hat{f}_l(Z^q_{(N_q)}) | f_l(Z^q_{(1)}) \leq f_l(Z^q_{(2)}) \leq \cdots \leq f_l(Z^q_{(N_q)})\right)\right) \\ &\geq \prod_{q=1}^{k}\left(1-m\frac{\alpha}{B^q}\right), \end{aligned} \tag{15}$$

and then apply Bernoulli's inequality repeatedly to yield the final result. □

Theorem 2 holds in the noisy condition with (A1) assumption if the orderings of function values of each objective at iteration 1 to k are correct. Therefore, the quality of solution with noise is quantified as Corollary 5, which is the noisy version of Theorem 2.

Corollary 5 Consider the K^{th} iteration of MOPBnB on $(\mathscr{P})$ where (A1) is assumed. The intersection of the approximating set and $L(\delta,S)$ is non-empty with

$$P\left(\left(S\setminus\bigcup_{k=1}^{K}\sigma_p^k\right)\bigcap L(\delta,S) \neq \phi\right) \geq (1-\alpha)(1-m\alpha). \tag{16}$$

Under the noisy condition, Corollary 5 indicates that the intersection of the approximated Pareto optimal set Σ_k with the target set $L(\delta,S)$ is non-empty with at least $(1-\alpha)(1-m\alpha)$ probability.

In order to evaluate the incremental improvement of the algorithm, we consider a metric defined by Zabinsky et al. (2011), the *concentration* of the retained unpruned region. The concentration considers the proportion of desirable region in the remaining unpruned region, and is defined as

$$\mathscr{C}_k = \frac{v(\{S\setminus\cup_{i=1}^{k}\sigma_p^i\}\cap L(\delta,S))}{v(S\setminus\cup_{i=1}^{k}\sigma_p^i)}, \tag{17}$$

where $v()$ is the volume of a region of solutions. When a dimension is associated with a discrete variable, the volume is calculated as counting the elements in the region.

Theorem 6 Assuming the algorithm progresses to K^{th} iteration and the best distance of the sampled points has not reached the desirable set, i.e., $D(X_{NS_k,(1)}) > y(\delta, S)$ for $k = 1, \ldots, K-1$, then

$$P(\mathscr{C}_{k+1} \geq \mathscr{C}_k) \geq (1-\alpha)(1-m\alpha). \tag{18}$$

Proof. See Appendix A. □

4 Numerical Experiments

Two test functions, he Fonseca and Fleming function and the Kursawe function (Deb 2001), are used in this section to illustrate the performance of MOPBnB, both test functions have two objective functions and are box-constrained. The Fonseca and Fleming function has a two-dimensional feasible region S, so we can visually illustrate the algorithm on the solution space. The Kursawe function is in three dimensional space, so we only illustrate the efficient frontier.

- Fonseca and Fleming function ($n = 2, -4 \leq x_i \leq 4, i = 1, \ldots, n$)

$$\text{Minimize} = \begin{cases} g_1(x) = 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i - \frac{1}{\sqrt{n}}\right)^2\right) \\ g_2(x) = 1 - \exp\left(-\sum_{i=1}^{n}\left(x_i + \frac{1}{\sqrt{n}}\right)^2\right) \end{cases} \tag{19}$$

- Kursawe function ($n = 3$ and $-5 \leq x_i \leq 5, i = 1, \ldots, n$)

$$\text{Minimize} = \begin{cases} g_1(x) = \sum_{i=1}^{2}\left[-10\exp\left(-0.2\sqrt{x_i^2 + x_{i+1}^2}\right)\right] \\ g_2(x) = \sum_{i=1}^{3}\left[|x_i|^0.8 + 5\sin(x_i^3)\right] \end{cases} \tag{20}$$

The parameters were set as $\delta = 0.1$, $\alpha = 0.05$, and $B = 2$. The ε parameter was set at one percent of the longest Euclidean distance of S. For both test functions, we tested with no noise, $\xi_{x,l} = 0$, and with noise, $\xi_{x,l} = N(0, 0.3)$ for Fonseca and Fleming function and $\xi_{x,l} = N(0,1)$ for Kursawe function. The noise is additive, $g_l(x) + \xi_{x,l}$. The variability of the noise was determined by considering of the range of the efficient frontier's function value.

Figure 1 illustrates the efficient frontier and the approximated Pareto optimal set for the Fonseca and Fleming function. Panels (A) and (C) of Figure 1 are with no noise, and panels (B) and (D) are with noise. Panels (A) and (B) illustrate the efficient frontier in the objective function space and panels (C) and (D) illustrate the approximated Pareto optimal set in the solution space. In panel (A) of Figure 1, the stars (blue) represent the empirically estimated true efficient frontier and were obtained by performing grid search for 100x100 points. The dots (red) are the estimated objective function values from NS_k in the MOPBnB algorithm at the final iteration ($k = 14$), there are 1,589 non-dominated points in NS_k, out of 11,643 total sampled points. As illustrated, our approximation of the efficient frontier shows excellent closeness and spreading. Panel (B) of Figure 1 includes noise, $\xi_{x,l} = N(0, 0.3)$. The stars (blue) are the same as in panel (A), and the dots (red) are the estimated objective function values $\hat{f}_l$ from the MOPBnB algorithm, under the noisy condition. It is interesting that the red dots appear "better" than the blue stars, so we also plotted the true function values with no noise, f_l (blue circles) associated with the non-dominated points. The blue circles are actually overlapping or very close to the true efficient frontier. Furthermore, in the noisy condition, there were 22 non-dominated points in NS_k, out of 8,742 total sampled points.

Panel (C) of Figure 1 illustrates the progress of the MOPBnB algorithm in the solution space, with the final subregions approximating the Pareto optimal set. Each rectangle represents a subregion in the

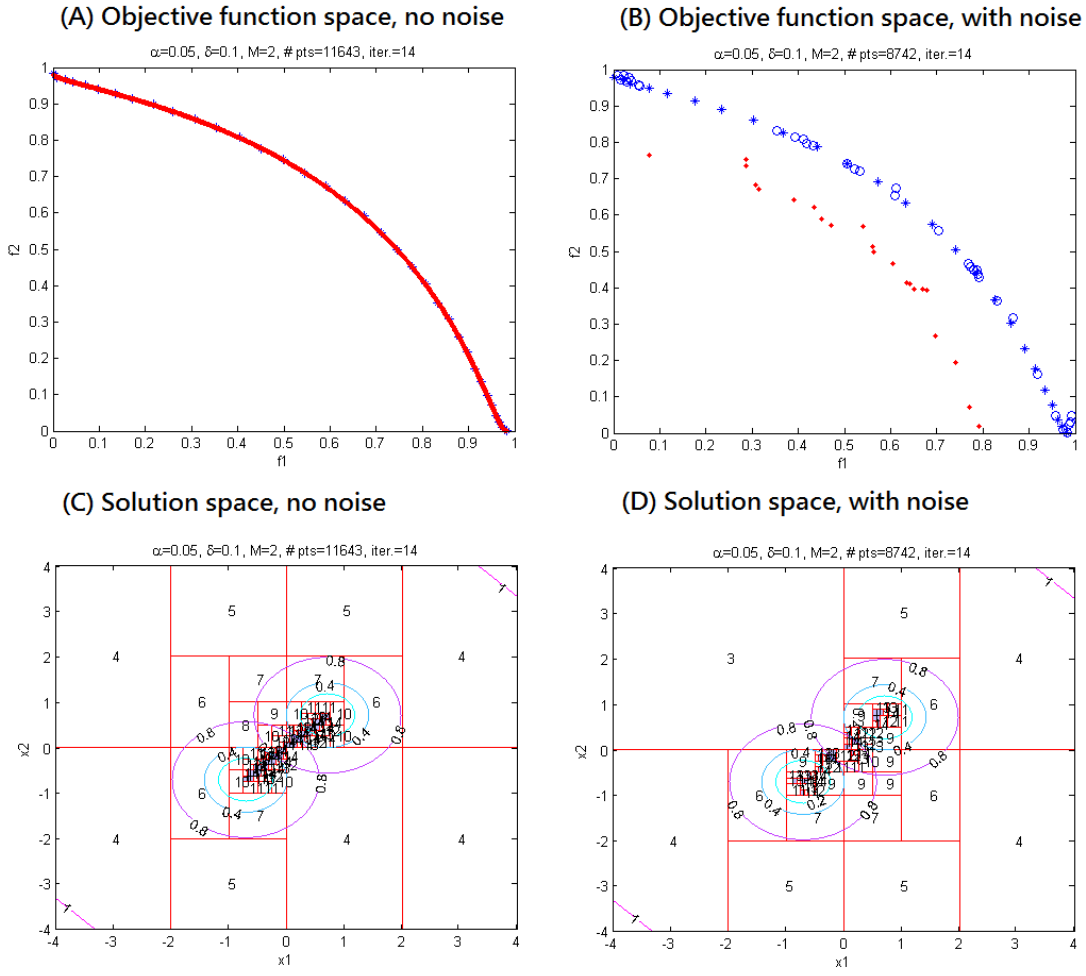

Figure 1: Fonseca and Fleming function. Panels (A) and (B) plot the two objective functions to illustrate the efficient frontier, (A) no noise, $\xi_x = 0$, and (B) with noise, $\xi_x = N(0,1)$. Panels (C) and (D) plot the approximated Pareto optimal set, (C) no noise, $\xi_x = 0$, and (D) with noise, $\xi_x = N(0,1)$.

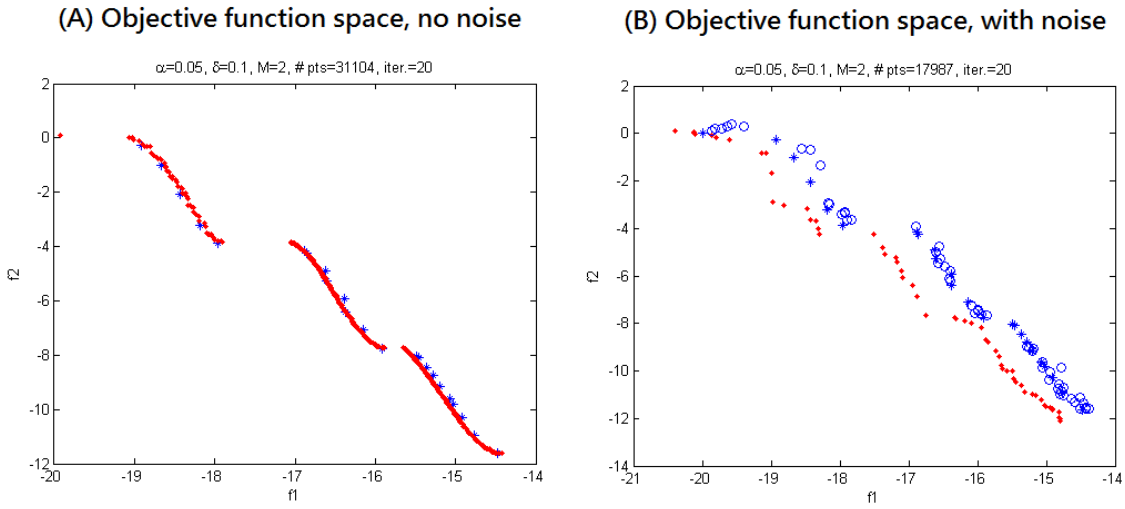

Figure 2: Kursawe function. Panels (A) and (B) plot the two objective functions to illustrate the efficient frontier, (A) no noise, $\xi_x = 0$ and (B) with noise, $\xi_x = N(0,1)$.

algorithm, where the number in a subregion represents the iteration that the subregion is pruned. Similarly, panel (D) in Figure 1 shows the results of Fonseca and Fleming function with $\xi_{x,l} = N(0, 0.3)$.

For the Kursawe function, only the approximate efficient frontiers (no noise and $N(0,1)$ noise) are shown in Figure 2, because the solution space in three dimension is difficult to visualize. The approximated efficient frontier illustrates a good approximation to the true efficient frontier with regard to both closeness and spread. Similarly, the noise version in panel (B) of Figure 2 shows the bias of the estimated efficient frontier (red dots), but the corresponding true function values (blue circles) estimate the efficient frontier well. For no noise, the algorithm found 415 non-dominated solutions out of 31,104 points sampled, and, with noise, the algorithm found 56 non-dominated solutions from total 17,967 samples.

Both test functions show that the influence of noise decreases the spread and the size of the approximated Pareto optimal set. A possible reason is that the MOPBnB is currently designed by only considering the closeness of the solution. The sample size and pruning step do not consider the spread and the potential loss of Pareto optimal set. Hence, a good future research direction is to incorporate the spread and the size of approximated Pareto optimal set to the algorithm.

5 Conclusion

We develop the MOPBnB algorithm to provide the approximated efficient frontier by the objective function values of the sample points in NS_k in the objective space and the approximated Pareto optimal set Σ_k in the solution space for multiple objective stochastic global optimization problems. The algorithm has the capability to work with mixed-integer/continuous variables. The framework of MOPBnB is to partition the solution space iteratively and determine whether each subregion should be pruned or further branched depending on the observed non-dominated samples. The algorithm provides the quality of the approximated Pareto optimal set Σ_k by considering the distance to the true efficient frontier in the objective space. We prove that, with probability $(1-\alpha)(1-m\alpha)$, the approximated Pareto optimal set intersects the relaxed Pareto set $L(\delta, S)$. We also prove that the concentration of "good" solutions in the retained region increases with iteration.

The potential future work to this algorithm includes two directions, deriving further quality for the current algorithm and developing a new version of MOPBnB with different metrics. In general, approximating the efficient frontier considers three criteria, closeness, spread and the number of points. MOPBnB is only driven by the distance, an indicator of closeness. Considering the other two metrics could provide more

comprehensive insights to the quality of MOPBnB. Another future direction is to redefine the sample size and partition scheme to develop a new version of the MOPBnB with the consideration of spread and the number of points.

ACKNOWLEDGMENTS

This work has been funded in part by the Department of Laboratory Medicine at Seattle Children's Hospital, and by NSF grant CMMI-1235484.

A Proof of Theorem 6

Proof. (Based on Wang (2011)) For any $k = 1, \ldots, K-1$, we rewrite the concentration of desired solutions $\mathscr{C}_{k+1}$ after the iteration $k+1$ as follows,

$$\frac{v(\{S \setminus \cup_{i=1}^{k+1} \sigma_p^i\} \cap L(\delta, S))}{v(S \setminus \cup_{i=1}^{k+1} \sigma_p^i)} = \frac{v(\{S \setminus \cup_{i=1}^{k} \sigma_p^i\} \cap L(\delta, S)) - v(\sigma_p^{k+1} \cap L(\delta, S))}{v(S \setminus \cup_{i=1}^{k} \sigma_p^i) - v(\sigma_p^{k+1})} \equiv \frac{A-B}{C-D}. \tag{21}$$

Thus (18) is equivalent to $P\left(\frac{A-B}{C-D} \geq \frac{A}{C}\right) \geq (1-\alpha)(1-m\alpha)$. Note that with $A > B, C > D$ and $A, B, C, D > 0$,

$$\frac{A-B}{C-D} \geq \frac{A}{C} \Leftrightarrow AC - BC \geq AC - AD \Leftrightarrow AD \geq BC \Leftrightarrow \frac{A}{C} \geq \frac{B}{D}. \tag{22}$$

For each iteration, if $||\sigma_p^k|| = 0$, then the proof is trivial. Therefore, consider $||\sigma_p^k|| \geq 1$ First, consider the $\frac{B}{D}$ part. Let D_p^k be the best performance associated to a sample point from all points in the pruned regions at iteration k, and let $D_{NS,p}^k$ be the distance performance associated to a sample from the non-dominated set which dominates the sample point associated to D_p^k considering all m objectives. From Theorem 3, under (A1) assumption, we know that the order will be correct with $1-\alpha_k$ probability for one objective. Hence, with $1-m\alpha_k$ probability, $D_{NS,p}^k < D_p^k$. Let σ_p^k be partitioned into the following two disjoint sets

$$\left\{x \in \sigma_p^k : D(x) < D_p^k\right\} \text{and} \left\{x \in \sigma_p^k : D(x) \geq D_p^k\right\}.$$

Also, $D_{NS,p}^k \geq D(X_{NS,(1)}) > y(\delta, S)$ by the assumption. Therefore, there is $1-m\alpha_k$ probability that

$$\left\{x \in \sigma_p^k : D(x) \geq D_p^k\right\} \bigcap L(\delta, S) = \phi.$$

Hence, with $1-m\alpha_k$ probability, we can say

$$\begin{aligned} v\left(\sigma_p^k \bigcap L(\delta, S)\right) &= v\left(\left\{x \in \sigma_p^k : D(x) < D_p^k\right\} \bigcap L(\delta, S)\right) + v\left(\left\{x \in \sigma_p^k : D(x) \geq D_p^k\right\} \bigcap L(\delta, S)\right) \\ &= v\left(\left\{x \in \sigma_p^k : D(x) < D_p^k\right\} \bigcap L(\delta, S)\right) \\ &\leq v\left(\left\{x \in \sigma_p^k : D(x) < D_p^k\right\}\right). \end{aligned} \tag{23}$$

By Theorem 1, it holds that

$$P(D_p^k < y(\delta, \sigma_p^k)) \geq 1 - \alpha_k. \tag{24}$$

Combining (23) and (24), with at least $(1-\alpha_k)(1-m\alpha_k)$ probability

$$v\left(\sigma_p^k \bigcap L(\delta, S)\right) \leq v\left(\left\{x \in \sigma_p^k : D(x) < y(\delta, \sigma_p^k)\right\}\right). \tag{25}$$

By the definition of quantile, we have

$$\frac{v\left(\left\{x \in \sigma_p^k : D(x) < y(\delta, \sigma_p^k)\right\}\right)}{v\left(\sigma_p^k\right)} \leq \delta. \tag{26}$$

Combining (25) and (26), we have at least $(1-\alpha_k)(1-m\alpha_k)$ probability that

$$\frac{B}{D} = \frac{v\left(\sigma_p^k \bigcap L(\delta, S)\right)}{v\left(\sigma_p^k\right)} \leq \delta. \tag{27}$$

Replacing k with $k+1$ since (27) is for any iteration, we have

$$\frac{B}{D} = \frac{v\left(\sigma_p^{k+1} \bigcap L(\delta, S)\right)}{v\left(\sigma_p^{k+1}\right)} \leq \delta. \tag{28}$$

For the $\frac{A}{C}$ part,because $S \bigcap L(\delta, S) = L(\delta, S)$ and $v(L(\delta, S)) \geq \delta v(S)$ by quantile definition, we can derive

$$\begin{aligned} v\left(\left\{S \setminus \cup_{i=1}^k \sigma_p^i\right\} \bigcap L(\delta, S)\right) &= v\left(S \bigcap L(\delta, S)\right) - \sum_{i=1}^k v\left(\sigma_p^i \bigcap L(\delta, S)\right) \\ &\geq \delta v(S) - \sum_{i=1}^k v\left(\sigma_p^i \bigcap L(\delta, S)\right) \\ &\geq \delta v(S) - \delta v(\sigma_p^1) - \sum_{i=2}^k v\left(\sigma_p^i \bigcap L(\delta, S)\right) \end{aligned}$$

with at least $(1-\alpha_k)(1-m\alpha_k)$ probability by (27). Then repeating the step for $i = 2, \ldots, k$, with at least $\prod_{i=1}^k (1-\alpha_i)(1-m\alpha_i)$ probability, the previous statement becomes

$$\geq \delta v(S) - \delta \sum_{i=1}^k v(\sigma_p^i) = \delta v\left(S \setminus \cup_{i=1}^k \sigma_p^i\right). \tag{29}$$

Dividing both sides by the volume term on the right hand side, we have, with at least $\prod_{i=1}^k 1-\alpha_i)(1-m\alpha_i)$ probability,

$$\frac{A}{C} = \frac{v\left(\left\{S \setminus \cup_{i=1}^k \sigma_p^i\right\} \bigcap L(\delta, S)\right)}{v\left(S \setminus \cup_{i=1}^k \sigma_p^i\right)} \geq \delta. \tag{30}$$

Combining (28) and (30), we have that $\frac{A}{C} \geq \frac{B}{D}$ with at least $\prod_{i=1}^{k+1} 1(-\alpha_i)(1-m\alpha_i)$ probability, which is strictly greater than $(1-\alpha)(1-m\alpha)$ by Bernoulli's inequality and the proof of (18) is completed. □

REFERENCES

Bechhofer, R. E., C. W. Dunnett, and M. Sobel. 1954. "A Two-Sample Multiple Decision Procedure for Ranking Means of Normal Populations with a Common Unknown Variance". *Biometrika* 41:170–176.

Bianchi, L., M. Dorigo, L. M. Gambardella, and W. J. Gutjahr. 2008. "A Survey on Metaheuristics for Stochastic Combinatorial Optimization". *Natural Computing* 8 (2): 239–287.

Deb, K. 2001. *Multi-objective Optimization Using Evolutionary Algorithms*. New York: John Wiley & Sons, Chichester.

Gibbons, J. D., I. Olkin, and M. Sobel. 1977. *Selecting and Ordering Populations: A New Statistical Methodology*. New York: Wiley.

Huang, H., and Z. B. Zabinsky. 2013. "Adaptive Probabilistic Branch and Bound with Confidence Intervals for Level Set Approximation". In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S. H. Kim, A. Tolk, R. Hill, and M. E. Kuhl. Washington D.C.: Institute of Electrical and Electronics Engineers, Inc.

Jones, D. F., S. K. Mirrazavi, and M. Tamiz. 2002. "Multi-objective Meta-heuristics: An Overview of the Current State-of-the-art". *European Journal of Operational Research* 137 (1): 1–9.

Kung, H. T., F. Luccio, and F. P. Preparata. 1975. "On Finding the Maxima of a Set of Vectors". *Journal of the ACM* 22:469–476.

Lee, L. H., E. P. Chew, S. Teng, and Y. Chen. 2008. "Multi-objective Simulation-based Evolutionary Algorithm for an Aircraft Spare Parts Allocation Problem". *European Journal of Operational Research* 189 (2): 476–491.

Lee, L. H., E. P. Chew, S. Teng, and D. Goldsman. 2004. "Optimal Computing Budget Allocation for Multi-objective Simulation Models". In *Simulation Conference, 2004. Proceedings of the 2004 Winter*, Volume 1, 594.

Lee, L. H., E. P. Chew, S. Teng, and D. Goldsman. 2010. "Finding the Non-dominated Pareto Set for Multi-objective Simulation Models". *IIE Transactions* 42 (9): 656–674.

Lin, R.-C., M. Y. Sir, and K. S. Pasupathy. 2013. "Multi-objective Simulation Optimization Using Data Envelopment Analysis and Genetic Algorithm: Specific Application to Determining Optimal Resource Levels in Surgical Services". *Omega* 41 (5): 881–892.

Tekin, E., and I. Sabuncuoglu. 2004. "Simulation Optimization: A Comprehensive Review on Theory and Applications". *IIE Transactions* 36 (11): 1067–1081.

Wang, W. 2011. "Adaptive Random Search for Noisy and Global Optimization". *Ph. D. thesis, University of Washington*.

Zabinsky, Z. B., W. Wang, Y. Prasetio, A. Ghate, and J. Yen. 2011. "Adaptive Probabilistic Branch and Bound for Level Set Approximation". In *Proceedings of the 2011 Winter Simulation Conference*, edited by S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, 46–57. Phoenix, Arizona: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

HAO HUANG is a PhD candidate in the Department of Industrial and Systems Engineering at the University of Washington. His research interests include simulation optimization and healthcare applications. His e-mail is haoh7493@uw.edu.

ZELDA B. ZABINSKY is a Professor in the Department of Industrial and Systems Engineering at the University of Washington, with adjunct appointments in the departments of Electrical Engineering, Mechanical Engineering, and Civil and Environmental Engineering. She is an IIE Fellow, and active in INFORMS, serving as General Chair for the INFORMS Annual Meeting held in Seattle in 2007. She has published numerous papers in the areas of global optimization, algorithm complexity, and optimal design of composite structures. Her book, *Stochastic Adaptive Search in Global Optimization*, describes research on theory and practice of algorithms useful for solving problems with multimodal objective functions in high dimension. She received an Erskine Fellowship from the University of Canterbury, Christchurch, New Zealand to collaborate internationally on global optimization methods. Professor Zabinsky's research interests are in global optimization under uncertainty for complex systems. Her email address is zelda@u.washington.edu.