

# Simulation-Based Optimization for Convex Functions Over Discrete Sets

Eunji Lim

Correspondence: Department of Decision Sciences and Marketing, School of Business, Adelphi University, Garden City, NY 11530, USA

Received: July 11, 2021 Accepted: August 10, 2021 Online Published: August 16, 2021

doi:10.5539/ijsp.v10n5p31 URL: <https://doi.org/10.5539/ijsp.v10n5p31>

## Abstract

We propose a new iterative algorithm for finding a minimum point of  $f_* : X \subset \mathbb{R}^d \rightarrow \mathbb{R}$ , when  $f_*$  is known to be convex, but only noisy observations of  $f_*(\mathbf{x})$  are available at  $\mathbf{x} \in X$  for a finite set  $X$ . At each iteration of the proposed algorithm, we estimate the probability of each point  $\mathbf{x} \in X$  being a minimum point of  $f_*$  using the fact that  $f_*$  is convex, and sample  $r$  points from  $X$  according to these probabilities. We then make observations at the sampled points and use these observations to update the probability of each point  $\mathbf{x} \in X$  being a minimum point of  $f_*$ . Therefore, the proposed algorithm not only estimates the minimum point of  $f_*$  but also provides the probability of each point in  $X$  being a minimum point of  $f_*$ . Numerical results indicate the proposed algorithm converges to a minimum point of  $f_*$  as the number of iterations increases and shows fast convergence, especially in the early stage of the iterations.

**Keywords:** simulation-based optimization, convex optimization, probability of correct selection

## 1. Introduction

We consider the problem of finding a minimum point  $\mathbf{x}_*$  of a convex function  $f_* : X \subset \mathbb{R}^d \rightarrow \mathbb{R}$  when  $f_*(\mathbf{x})$  cannot be evaluated exactly but can be observed only with noise at each  $\mathbf{x} \in X$ , possibly via stochastic simulation. Specifically, we assume we can obtain a noisy observation  $Y(\mathbf{x})$  of  $f_*(\mathbf{x})$ , satisfying  $Y(\mathbf{x}) = f_*(\mathbf{x}) + \varepsilon(\mathbf{x})$  for some random variable  $\varepsilon(\mathbf{x})$  with a mean of zero and a variance of  $\sigma^2(\mathbf{x})$  at each  $\mathbf{x} \in X$ . Our focus in this paper is on the case where  $X$  is a finite set, i.e.,  $X = \{\mathbf{x}^1, \dots, \mathbf{x}^l\} \subset \mathbb{R}^d$ . Thus, we say  $f_* : X \subset \mathbb{R}^d \rightarrow \mathbb{R}$  is convex if there exists a convex function  $\bar{f}_* : \mathbb{R}^d \rightarrow \mathbb{R}$  satisfying  $f_*(\mathbf{x}^i) = \bar{f}_*(\mathbf{x}^i)$  for  $1 \leq i \leq l$ . Such convexity arises in many applications. For example, the expected loss function in the classical newsvendor problem is proved to be convex in the order quantity, which takes on values in a discrete set (Porteus, 1990). The expected completion time of the  $i$ th customer, for  $i = 1, 2, \dots$ , in a tandem queueing system is proved to be convex in the service rate (Shanthikumar & Yao 1989).

In this paper, we propose an iterative algorithm for finding a minimum point  $\mathbf{x}_*$  of  $f_*$ . At each iteration of the proposed algorithm, we sample  $r$  points from  $X$ , obtain  $m$  observations from each sampled point, and use these observations to decide which points to sample in the next iteration. The key feature of the proposed algorithm is that at each iteration, we compute the probability of  $\mathbf{x}$  being a minimum point of  $f_*$  for each  $\mathbf{x} \in X$  using the fact that  $f_*$  is convex and use these probabilities to sample points from  $X$  in the next iteration.

More specifically, our algorithm proceeds as follows. Initially, we assume every point in  $X$  is equally likely to be a minimum point of  $f_*$ . Thus, we sample  $r$  points, say  $\{\mathbf{x}_1, \dots, \mathbf{x}_r\}$ , uniformly from  $X$ . We then obtain  $m$  independent realizations of  $Y(\mathbf{x}_i)$  for  $i = 1, 2, \dots, r$ . We will denote the average and variance of these  $m$  copies of  $Y(\mathbf{x}_i)$  by  $\bar{Y}(\mathbf{x}_i)$  and  $\hat{\sigma}^2(\mathbf{x}_i)$  for  $i = 1, 2, \dots, r$ , respectively. The next step is to compute the probability of  $\mathbf{x}$  being a minimum point of  $f_*$  at each  $\mathbf{x} \in X$ . To compute this, we consider the following quantity:

$$q(\mathbf{x}) = \min_{f \in \mathcal{F}_{\mathbf{x}}} \sum_{i=1}^r \frac{m}{\hat{\sigma}^2(\mathbf{x}_i)} (\bar{Y}(\mathbf{x}_i) - f(\mathbf{x}_i))^2,$$

where  $\mathcal{F}_{\mathbf{x}}$  is the set of convex functions  $f : X \subset \mathbb{R}^d \rightarrow \mathbb{R}$  that have  $\mathbf{x}$  as a minimum point. If  $\mathbf{x}$  is indeed a minimum point of  $f_*$ , then

$$q(\mathbf{x}) = \min_{f \in \mathcal{F}_{\mathbf{x}}} \sum_{i=1}^r \frac{m}{\hat{\sigma}^2(\mathbf{x}_i)} (\bar{Y}(\mathbf{x}_i) - f(\mathbf{x}_i))^2 \leq \sum_{i=1}^r \frac{m}{\hat{\sigma}^2(\mathbf{x}_i)} (\bar{Y}(\mathbf{x}_i) - f_*(\mathbf{x}_i))^2 = \sum_{i=1}^r \frac{m}{\hat{\sigma}^2(\mathbf{x}_i)} \bar{\varepsilon}(\mathbf{x}_i)^2,$$

where  $\bar{\varepsilon}(\mathbf{x}_i)$  is the average of  $m$  independent copies of  $\varepsilon(\mathbf{x}_i)$ . We note that the rightmost term in the above equation,

$$\sum_{j=1}^r \frac{m}{\hat{\sigma}^2(\mathbf{x}_j)} \bar{\varepsilon}(\mathbf{x}_j)^2$$

converges weakly to a  $\chi^2$ -distribution with  $r$  degrees of freedom as  $m$  increases. Therefore, we can approximate the probability of  $\mathbf{x}$  being a minimum point of  $f_*$  by the probability of observing a value that is as extreme as  $q(\mathbf{x})$  from a  $\chi^2$ -distribution with  $r$  degrees of freedom, which in turn is  $\mathbb{P}(\chi^2(r) \geq q(\mathbf{x}))$ , where  $\chi^2(r)$  represents the  $\chi^2$ -distribution with  $r$  degrees of freedom. The remaining question is how to compute  $q(\mathbf{x})$ . Proposition 1 of this paper shows that  $q(\mathbf{x})$  can be computed by solving a quadratic program. With the probability of  $\mathbf{x}$  being a minimum point of  $f_*$  at each  $\mathbf{x} \in X$ , our next iteration proceeds by sampling another set of  $r$  points from  $X$  proportionately to these probabilities. A similar procedure is repeated in each iteration. At each iteration, our estimate of the minimum point of  $f_*$  is the point that achieves the minimum value of the average of all observations made at each point sampled so far.

Simulation-based optimization over discrete sets has received a great deal of attention in the research community. On the one hand, when one assumes no structure on  $f_*$ , random search type methods are the most widely used; see Glover (1989) for the tabu search, Liepins and Hilliard (1989) for the generic algorithm, Chen (1996) and Kim and Nelson (2006) for the ranking and selection method, Yan and Mukai (1992) for the stochastic ruler method, Alrefaei and Andradóttir (1999) for the simulated annealing method, Gong, Ho and Zhai (1999) for the stochastic comparison method, Shi and Ólafsson (2000) for the nested partitions method, and Hong and Nelson (2006) for convergent optimization via most-promising-area stochastic search (COMPASS).

On the other hand, when one has a priori knowledge of the shape of  $f_*$ , such as convexity, one can exploit it to design a more efficient algorithm for finding a minimum point of  $f_*$ . A number of local search methods have been proposed, and they can be applied to the case where  $f_*$  is known to be convex; see Gerencsér et al. (1999) and Wang et al. (2013). However, most of them are gradient-based methods, which use a gradient estimate to update their estimate of the minimum point of  $f_*$  at each iteration. The performance of gradient-based methods is, in general, highly sensitive to the choice of the step-size sequence, and how to choose the step-size sequence remains challenging. Another drawback of a gradient-based method is that at each iteration, it uses the most recently sampled point and its gradient estimate only, ignoring all the previously sampled points and the observations made at these points.

In this paper, we take a different view, beginning with the following two observations.

1. When  $f_*$  is known to be convex, it is desirable to utilize all the previously sampled points and the observations made at these points because a point that is far from  $\mathbf{x}_*$  can still add information on  $\mathbf{x}_*$ . For example, suppose we wish to estimate a convex regression function  $g_* : \mathbb{R}^d \rightarrow \mathbb{R}$ . By sampling  $n$  points from the domain randomly, making observations at the sampled points, and fitting a convex function to the data set, one can successfully estimate  $g_*(\mathbf{x})$  at  $\mathbf{x} \in X$  consistently as  $n$  increases to infinity; see Seijo & Sen (2011). This shows that each observation in the data set makes a contribution to the estimation of  $g_*(\mathbf{x})$ , even if it is located far from  $\mathbf{x}$ .
2. Most existing methods generate an estimate of a minimum point of  $f_*$  at each iteration, but they do not provide the probability of their choice being a minimum of  $f_*$ . It is nonetheless desirable to do so because the probability will give a sense of how confident we can be about the estimate.

In our proposed algorithm, we make use of all the previously sampled points and all the observations made at these points. We also provide the probability of each point in  $X$  being a minimum point of  $f_*$ . To the best of our knowledge, this paper is the first to achieve this latter result.

The rest of this paper is organized as follows. In Section 2, we introduce some notation and definitions. In Section 3, we describe the proposed algorithm in greater detail. In Section 4, we present the numerical performance of the proposed algorithm in two settings: (1) a stylized model in which  $f_*$  is given by a quadratic equation and (2) a more realistic model in which  $f_*$  is the expected completion time of the 10th customer in a tandem queueing system. We provide concluding remarks and suggest future research directions in Section 5.

## 2. Notation and Definitions

For  $\mathbf{x} = (x_1, x_2, \dots, x_d) \in \mathbb{R}^d$ ,  $\mathbf{x}^T$  and  $\|\mathbf{x}\|$  denote the transpose of  $\mathbf{x}$  and  $(x_1^2 + \dots + x_d^2)^{1/2}$ , respectively. For any set  $S$ ,  $|S|$  denotes the number of elements in  $S$ . For a function  $g : \mathbb{R}^m \rightarrow \mathbb{R}$ , a vector  $\xi \in \mathbb{R}^m$  is said to be a subgradient of  $g$  at  $x \in \mathbb{R}^m$  if  $g(y) \geq g(x) + \xi^T(y - x)$  for all  $y \in \mathbb{R}^m$ .

### 3. Proposed Algorithm

In this section, we give a full description of the proposed algorithm. In each iteration of the proposed algorithm, we will sample  $r$  points from  $X$  according to the probabilities  $(p(\mathbf{x}) : \mathbf{x} \in X)$ , where  $p(\mathbf{x})$  is an estimate of the probability of  $\mathbf{x}$  being a minimum point of  $f_*$ , and take  $m$  observations from each of the  $r$  sampled points. Thus, the total number of observations made at  $\mathbf{x} \in X$  up to iteration  $k$ , which we will denote by  $N_k(\mathbf{x})$ , is updated at each iteration. We will also denote the average and variance of all the observations made at  $\mathbf{x} \in X$  up to iteration  $k$  by  $\bar{Y}_k(\mathbf{x})$  and  $\hat{\sigma}_k^2(\mathbf{x})$ , respectively. In addition,  $\mathcal{S}_k$  is the set of all the points in  $X$  sampled up to iteration  $k$ . At iteration  $k$ , our estimate of  $\mathbf{x}_*$  is the point in  $\mathcal{S}_k$  that has the minimum value of  $\bar{Y}_k(\mathbf{x})$  and will be denoted by  $\hat{\mathbf{x}}_k^*$ .

Below is the detailed description of the proposed algorithm.

#### Proposed Algorithm

*Step 0:* Set the iteration count  $k = 0$ . Let  $(p(\mathbf{x}) : \mathbf{x} \in X)$  be the uniform distribution over  $X$ , i.e.,  $p(\mathbf{x}) = 1/|X|$  for each  $\mathbf{x} \in X$ . Set  $\mathcal{S}_k = \emptyset$  and  $N_k(\mathbf{x}) = 0$  for each  $\mathbf{x} \in X$ .

*Step 1:* Set  $k = k + 1$ . Sample  $r$  points, say  $\{\mathbf{x}_{k1}, \mathbf{x}_{k2}, \dots, \mathbf{x}_{kr}\}$ , from  $X$  proportionately to  $(p(\mathbf{x}) : \mathbf{x} \in X)$  with replacement, i.e.,  $\mathbf{x} \in X$  is sampled with probability  $p(\mathbf{x}) / \sum_{\mathbf{z} \in X} p(\mathbf{z})$ . Take  $m$  observations from each of  $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \dots, \mathbf{x}_{kr}$  and update  $N_k(\mathbf{x})$ ,  $\bar{Y}_k(\mathbf{x})$ , and  $\hat{\sigma}_k^2(\mathbf{x})$  for  $\mathbf{x} \in X$ . Set  $\mathcal{S}_k = \mathcal{S}_{k-1} \cup \{\mathbf{x}_{k1}, \mathbf{x}_{k2}, \dots, \mathbf{x}_{kr}\}$ .

*Step 2:* For each  $\mathbf{x} \in X$ , compute  $q(\mathbf{x})$ , the probability of  $\mathbf{x}$  being a minimum point of  $f_*$ . To compute  $q(\mathbf{x})$ , one must compute  $q(\mathbf{x})$

$$q(\mathbf{x}) \triangleq \min_{f \in \mathcal{F}_x} \sum_{\mathbf{z} \in \mathcal{S}_k} \frac{N_k(\mathbf{z})}{\hat{\sigma}_k^2(\mathbf{z})} (\bar{Y}_k(\mathbf{z}) - f(\mathbf{z}))^2, \tag{1}$$

where  $\mathcal{F}_x$  is the set of convex functions  $f : X \subset \mathbb{R}^d \rightarrow \mathbb{R}$  having  $\mathbf{x}$  as a minimum point. One can compute  $q(\mathbf{x})$  by solving the quadratic program in Proposition 1, which can be found below. Once  $q(\mathbf{x})$  is computed, we set  $p(\mathbf{x}) = \mathbb{P}(\chi^2(|\mathcal{S}_k|) \geq q(\mathbf{x}))$ , where  $\chi^2(|\mathcal{S}_k|)$  is the  $\chi^2$ -distribution with  $|\mathcal{S}_k|$  degrees of freedom and  $|\mathcal{S}_k|$  is the total number of points in  $X$  that have been sampled up to iteration  $k$ .

*Step 3:* Set  $\hat{\mathbf{x}}_k^* = \arg \min_{\mathbf{x} \in \mathcal{S}_k} \bar{Y}_k(\mathbf{x})$ . Go to Step 1.

One missing piece of information in the proposed algorithm is how to compute  $q(\mathbf{x})$ . The following proposition suggests  $q(\mathbf{x})$  can be computed by solving a quadratic program.

**Proposition 1.** At iteration  $k$ , assume  $N_k(\mathbf{x})$ ,  $\bar{Y}_k(\mathbf{x})$ , and  $\hat{\sigma}_k^2(\mathbf{x})$  are given for all  $\mathbf{x} \in \mathcal{S}_k$ .

For  $\mathbf{x} \in \mathcal{S}_k$ , define  $\bar{q}(\mathbf{x})$  by the optimal objective value of the following convex quadratic program:

$$\begin{aligned} \bar{q}(\mathbf{x}) \triangleq & \text{minimize} && \sum_{\mathbf{z} \in \mathcal{S}_k} \frac{N_k(\mathbf{z})}{\hat{\sigma}_k^2(\mathbf{z})} (\bar{Y}_k(\mathbf{z}) - g_{\mathbf{z}})^2 \\ & \text{subject to} && g_{\mathbf{z}} + \xi_{\mathbf{z}}^T (\mathbf{w} - \mathbf{z}) \leq g_{\mathbf{w}}, && \mathbf{z}, \mathbf{w} \in \mathcal{S}_k \\ & && g_{\mathbf{x}} \leq g_{\mathbf{z}}, && \mathbf{z} \in \mathcal{S}_k \end{aligned} \tag{2}$$

in the decision variables  $g_{\mathbf{z}} \in \mathbb{R}$  and  $\xi_{\mathbf{z}} \in \mathbb{R}^d$  for all  $\mathbf{z} \in \mathcal{S}_k$ . In (2),  $g_{\mathbf{z}}$  represents  $g(\mathbf{z})$  for  $g \in \mathcal{F}_x$ .

For  $\mathbf{x} \notin \mathcal{S}_k$ , define  $\bar{q}(\mathbf{x})$  by the optimal objective value of the following convex quadratic program:

$$\begin{aligned} \bar{q}(\mathbf{x}) \triangleq & \text{minimize} && \sum_{\mathbf{z} \in \mathcal{S}_k} \frac{N_k(\mathbf{z})}{\hat{\sigma}_k^2(\mathbf{z})} (\bar{Y}_k(\mathbf{z}) - g_{\mathbf{z}})^2 \\ & \text{subject to} && g_{\mathbf{z}} + \xi_{\mathbf{z}}^T (\mathbf{w} - \mathbf{z}) \leq g_{\mathbf{w}}, && \mathbf{z}, \mathbf{w} \in \mathcal{S}_k \\ & && g_{\mathbf{x}} + \xi_{\mathbf{x}}^T (\mathbf{w} - \mathbf{x}) \leq g_{\mathbf{w}}, && \mathbf{w} \in \mathcal{S}_k \\ & && g_{\mathbf{z}} + \xi_{\mathbf{z}}^T (\mathbf{x} - \mathbf{z}) \leq g_{\mathbf{x}}, && \mathbf{z} \in \mathcal{S}_k \\ & && g_{\mathbf{x}} \leq g_{\mathbf{z}}, && \mathbf{z} \in \mathcal{S}_k \end{aligned} \tag{3}$$

in the decision variables  $g_{\mathbf{z}} \in \mathbb{R}$  and  $\xi_{\mathbf{z}} \in \mathbb{R}^d$  for all  $\mathbf{z} \in \mathcal{S}_k$ ,  $g_{\mathbf{x}} \in \mathbb{R}$ , and  $\xi_{\mathbf{x}} \in \mathbb{R}^d$ . In (3),  $g_{\mathbf{z}}$  and  $g_{\mathbf{x}}$  represent  $g(\mathbf{z})$  and  $g(\mathbf{x})$ , respectively, for  $g \in \mathcal{F}_x$ .

Then, we have  $q(\mathbf{x}) = \bar{q}(\mathbf{x})$ , where  $q(\mathbf{x})$  is defined in (1).

Table 1. The 95% confidence interval of  $\mathbb{E}[\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|]$  based on 100 copies of  $\hat{\mathbf{x}}_k^*$  when  $f_*(\mathbf{x}) = (\mathbf{x} - 50)^2$  for  $\mathbf{x} \in \{1, 2, \dots, 100\}$

Iteration	95% Confidence Interval of $\mathbb{E}[\ \hat{\mathbf{x}}_k^* - \mathbf{x}_*\ ]$
1	$5.47 \pm 0.91$
2	$1.69 \pm 0.23$
3	$1.50 \pm 0.21$
4	$1.37 \pm 0.20$
5	$1.18 \pm 0.20$
6	$1.02 \pm 0.15$
7	$1.07 \pm 0.15$
8	$0.98 \pm 0.16$
9	$0.89 \pm 0.15$
10	$0.73 \pm 0.14$
15	$0.64 \pm 0.12$

*Proof.* First, we focus on the case where  $\mathbf{x} \in \mathcal{S}_k$ . It suffices to show that (i) for any  $f \in \mathcal{F}_x$ , there exist  $g_z \in \mathbb{R}$  and  $\xi_z \in \mathbb{R}^d$  for all  $\mathbf{z} \in \mathcal{S}_k$  satisfying the constraints of (2), and (ii) for any  $g_z \in \mathbb{R}$  and  $\xi_z \in \mathbb{R}^d$ ,  $\mathbf{z} \in \mathcal{S}_k$ , satisfying the constraints of (2), there exists a function  $f \in \mathcal{F}_x$ . To prove (i), let  $f \in \mathcal{F}_x$ . Since  $f$  is convex, there exist subgradients of  $f$  at  $\mathbf{z} \in \mathcal{S}_k$ . Let  $\xi_z$  be a subgradient of  $f$  at  $\mathbf{z} \in \mathcal{S}_k$ . Also, let  $g_z = f(\mathbf{z})$  for  $\mathbf{z} \in \mathcal{S}_k$ . Then  $g_z$  and  $\xi_z$ ,  $\mathbf{z} \in \mathcal{S}_k$ , satisfy the constraints of (2); see, for example, Boyd and Vandenberghe (2004), p. 338. Conversely, let  $g_z$  and  $\xi_z$ ,  $\mathbf{z} \in \mathcal{S}_k$ , satisfy the constraints of (2). Define  $f : X \rightarrow \mathbb{R}$  by

$$f(\mathbf{x}) = \max_{\mathbf{z} \in \mathcal{S}_k} \{ \max\{g_z + \xi_z^T(\mathbf{x} - \mathbf{z}), g_x\} \}$$

for  $\mathbf{x} \in X$ . Then  $f \in \mathcal{F}_x$ . In the case where  $\mathbf{x} \notin \mathcal{S}_k$ , the proof follows in a similar fashion. □

#### 4. Numerical Results

In this section, we investigate the performance of the proposed algorithm in two settings: (1) a one-dimensional case where  $f_*(\mathbf{x}) = (\mathbf{x} - 50)^2$  for  $\mathbf{x} \in \{1, 2, \dots, 100\}$  and (2) a two-dimensional case where  $f_*$  is the expected completion time of the 10th customer in a tandem queueing system with two servers connected in series.

##### 4.1 One-Dimensional Case Where $d = 1$

We consider the case where  $X = \{1, 2, \dots, 100\}$ ,  $f_*(\mathbf{x}) = (\mathbf{x} - 50)^2$  for  $x \in X$ ,  $Y(\mathbf{x}) = f_*(\mathbf{x}) + \varepsilon(\mathbf{x})$  for  $\mathbf{x} \in X$ , and  $\varepsilon(\mathbf{x})$  follows a normal distribution with a mean of 0 and a standard deviation of 30. It should be noted that  $\mathbf{x}_* = 50$ . We applied the proposed algorithm with  $r = m = 10$ .

Figure 1 illustrates how  $\{p(\mathbf{x}) : \mathbf{x} \in X\}$  changes as the iteration progresses in a particular instance. One can observe that after the first iteration, the proposed algorithm successfully screened out 75% of the points in  $X$ . In the first iteration, we sampled 10 points from  $X$  randomly. Thus, the proposed algorithm screened out 75% of the points in  $X$  correctly by sampling only 10% of the points in  $X$ , using the fact that  $f_*$  is convex. In the second and third iteration, the proposed algorithm screened out 85% and 90%, respectively, of the points in  $X$ .

We computed  $\hat{\mathbf{x}}_k^*$  and  $\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|$  for  $k = 1, 2, \dots, 15$  and repeated this procedure 100 times independently. We denoted the mean and standard deviation of the 100 copies of  $\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|$  by  $\hat{m}_1$  and  $\hat{s}_1$ , respectively. Then  $\hat{m}_1 \pm (1.96)\hat{s}_1/\sqrt{100}$  represents the 95% confidence interval of  $\mathbb{E}[\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|]$  based on the 100 independent copies. In Table 1, we report the 95% confidence intervals for a range of the iteration count. The results show that the proposed algorithm converges to  $\mathbf{x}_*$  as the iteration increases, and the convergence is especially rapid in the early stage of the iterations.

##### 4.2 Two-Dimensional Case Where $d = 2$

We consider a tandem queueing network with two servers connected in series. The external arrival process to the first server follows a Poisson process with a rate of 100 customers an hour. A customer departing from the first server joins the queue for the second server. There are no external arrivals to the second server. A customer departing from the second server leaves the system. The variables for this example are the number of resources allocated to the first and second servers, say  $x^1$  and  $x^2$ . Thus,  $x^1$  and  $x^2$  take discrete values. We assume we have a total of 20 resources available, so  $X = \{(x^1, x^2) \in \mathbb{Z}^2 : x^1 + x^2 = 20, x^1 \geq 1, x^2 \geq 1\}$ . The service times for the first and second servers follow exponential distributions with rates of  $4x^1$  and  $10x^2$ , respectively, for  $(x^1, x^2) \in X$ . There is a buffer of infinite capacity in front of each server. Each server serves customers on a first in/first out basis. We assume the system is empty at time 0. For  $\mathbf{x} \in X$ ,

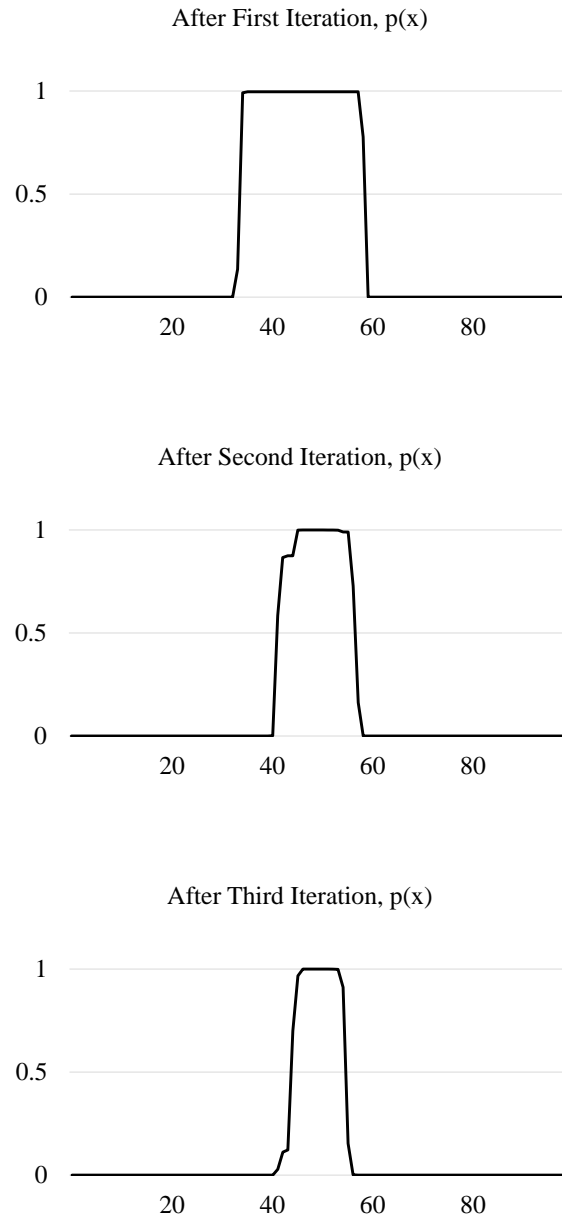


Figure 1. In all graphs, the  $x$  axis is  $\mathbf{x} \in X$  and the  $y$  axis is  $p(\mathbf{x})$ . The graphs on top, in the middle, and at the bottom represent  $p(\mathbf{x})$  after the first, second, and third iterations, respectively

Table 2. The 95% confidence interval of  $\mathbb{E}[\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|]$  based on 100 copies of  $\hat{\mathbf{x}}_k^*$  in the tandem queueing system

Iteration	95% Confidence Interval of $\mathbb{E}[\ \hat{\mathbf{x}}_k^* - \mathbf{x}_*\ ]$
1	$1.88 \pm 0.37$
2	$0.61 \pm 0.16$
3	$0.17 \pm 0.09$
4	$0.03 \pm 0.04$
5	$0.01 \pm 0.03$
6	$0.00 \pm 0.00$
7	$0.00 \pm 0.00$

$f_*(\mathbf{x})$  is the expected completion time of the 10th customer in the system. It should be noted that  $f_*(\mathbf{x})$  is proved to be convex in  $\mathbf{x}$ ; see Shanthikumar & Yao (1989).

We applied the proposed algorithm, with  $r = 5$  and  $m = 100$ , to obtain  $\hat{\mathbf{x}}_k^*$  and  $\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|$  for  $k = 1, 2, \dots, 7$ . We estimated  $\mathbf{x}_* = (13, 7)$  by generating 1,000,000 copies of  $Y(\mathbf{x})$  for each  $\mathbf{x} \in X$ , computing the average of the 1,000,000 copies at each  $\mathbf{x} \in X$ , and finding the minimum of these averages across  $\mathbf{x} \in X$ .

We computed  $\hat{\mathbf{x}}_k^*$  and  $\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|$  for  $k = 1, 2, \dots, 7$  and repeated this procedure 100 times independently. We denoted the mean and standard deviation of the 100 copies of  $\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|$  by  $\hat{m}_2$  and  $\hat{s}_2$ , respectively. Then  $\hat{m}_2 \pm (1.96)\hat{s}_2/\sqrt{100}$  represents the 95% confidence interval of  $\mathbb{E}[\|\hat{\mathbf{x}}_k^* - \mathbf{x}_*\|]$  based on the 100 independent copies. In Table 2, we report the 95% confidence intervals for  $k = 1, 2, \dots, 7$ . The results indicate the proposed algorithm converges to the true minimum point of  $f_*$  as the iteration progresses.

## 5. Conclusions

In this paper, we proposed a new algorithm for finding a minimum point  $\mathbf{x}_*$  of  $f_* : X \rightarrow \mathbb{R}$  when  $f_*$  is known to be convex, but  $f_*(\mathbf{x})$  for  $\mathbf{x} \in X$  cannot be evaluated exactly and must be estimated via simulation. At each iteration, the proposed algorithm not only suggests an estimate of  $\mathbf{x}_*$  but also computes the probability of each point  $\mathbf{x} \in X$  being a minimum point of  $f_*$ . The numerical results indicate the proposed algorithm shows fast convergence, especially in the early stage of the iterations.

There are multiple directions for future research. The proposed algorithm can be extended to the case where  $X$  is a continuous space or to the case where the observations are correlated. It also can be extended to the case where  $f_*$  has other shape characteristics such as unimodality or Lipschitz continuity.

## References

- Alrefaei, M. H., & Andradóttir, S. (1999). A simulated annealing algorithm with constant temperature for discrete stochastic optimization. *Management Science*, 45(5), 748-764. <https://doi.org/10.1287/mnsc.45.5.748>
- Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511804441>
- Chen, C. H. (1996). A lower bound for the correct subset-selection probability and its application to discrete-event system simulations. *IEEE Transactions on Automatic Control*, 41(8), 1227-1231. <https://doi.org/10.1109/9.533692>
- Gerencsér, L. S., Hill, D., & Vago, Z. (1999). Optimization over discrete sets via SPSA. In *Proceedings of the IEEE Conference on Decision and Control*, 1791-1795. <https://doi.org/10.1145/324138.324294>
- Glover, F. (1989). Tabu search—part I. *ORSA Journal on Computing*, 1(3), 190-206. <https://doi.org/10.1287/ijoc.1.3.190>
- Gong, W. B., Ho, Y. C., & Zhai, W. (1999). Stochastic comparison algorithm for discrete optimization with estimation. *SIAM Journal on Optimization*, 10(2), 384-404. <https://doi.org/10.1137/S1052623495290684>
- Hong, L. J., & Nelson, B. L. (2006). Discrete optimization via simulation using COMPASS. *Operations Research*, 54(1), 115-129. <https://doi.org/10.1287/opre.1050.0237>
- Kim, S. H., & Nelson, B. L. (2006). Selecting the best system. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in Operations Research and Management Science*, 13 (pp. 501–534). Elsevier. [https://doi.org/10.1016/S0927-0507\(06\)13017-0](https://doi.org/10.1016/S0927-0507(06)13017-0)
- Liepins, G. E., & Hilliard, M. R. (1989). Generic algorithms: Foundations and algorithms. *Annals of Operations Research*, 21, 31-58. <https://doi.org/10.1007/BF02022092>

- Porteus, E. L. (1990). Stochastic inventory theory. In D. P. Heyman & M. J. Sobel (Eds.), *Handbooks in Operations Research and Management Science 2* (pp. 605–652). Elsevier. [https://doi.org/10.1016/S0927-0507\(05\)80176-8](https://doi.org/10.1016/S0927-0507(05)80176-8)
- Shanthikumar J. G., & Yao, D. D. (1989). Second-order stochastic properties in queueing systems. In *Proceeding of the IEEE*, 77, 162-170. <https://doi.org/10.1109/5.21077>
- Shi, L., & Ólafsson, S. (2000). Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability*, 2, 271-291. <https://doi.org/10.1023/A:1010081212560>
- Wang, H., Pasupathy, R., & Schmeiser, B. W. (2013). Integer-ordered simulation optimization using  $r$ -spline: Retrospective search with piecewise-linear interpolation and neighborhood enumeration. *ACM Transactions on Modeling and Computer Simulation*, 23(3), 1-24. <https://doi.org/10.1145/2499913.2499916>

### Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).