

## TWO SIMULATED ANNEALING ALGORITHMS FOR NOISY OBJECTIVE FUNCTIONS

Andrei A. Prudius  
 Sigrún Andradóttir

School of Industrial and Systems Engineering  
 Georgia Institute of Technology  
 Atlanta, GA 30332-0205, U.S.A.

### ABSTRACT

We present two new variants of the simulated annealing algorithm (with a decreasing cooling schedule) that are designed for solving discrete simulation optimization problems. We also provide conditions under which our methods converge almost surely to the set of global optimal solutions, discuss the implications of our results for both transient and steady-state simulations, and provide some numerical results.

### 1 INTRODUCTION

Consider the following optimization problem

$$\max_{\theta \in \Theta} f(\theta) = \mathbb{E}[h_{\theta}(X_{\theta})], \quad (1)$$

where  $f : \Theta \rightarrow \mathbb{R}$  is the objective function,  $\Theta$  is the discrete feasible region, and for each  $\theta \in \Theta$ ,  $X_{\theta}$  is a random element in some space  $\mathcal{X}_{\theta}$  and  $h_{\theta} : \mathcal{X}_{\theta} \rightarrow \mathbb{R}$  is a deterministic function. Also let  $\Theta^*$  be the set of global optimal solutions to the problem (1); i.e.,  $\Theta^* = \{\theta \in \Theta : f(\theta) \geq f(\theta') \text{ for all } \theta' \in \Theta\}$ . We are interested in solving the problem (1) in situations where the objective function value  $f(\theta)$  at any  $\theta \in \Theta$  cannot be evaluated exactly, but needs to be estimated, for example via a “black-box” simulation procedure. This situation is common for discrete simulation optimization problems. Note that the form of  $f$  in (1) is general enough to include the settings where  $f$  is either a transient or a steady-state simulation performance measure. For some specific examples of optimization problems of the form (1), the interested reader is referred to Fu (2002).

In recent years, there has been an increasing interest in solving discrete simulation optimization problems. The recent work includes several new random search methods, such as the simulated annealing (SA) algorithms of Gelfand and Mitter (1989), Fox and Heine (1995), Gutjahr and Pflug (1996), and Alrefaei and Andradóttir (1999), the stochastic ruler methods of Yan and Mukai (1992) and Alrefaei and Andradóttir

(2001, 2005), the stochastic comparison methods of Andradóttir (1995, 1996, 1999) and Gong et al. (1999), the nested partitions methods of Shi and Ólafsson (2000) and Pichitlamken and Nelson (2003), the COMPASS method of Hong and Nelson (2005), and the BEES methods of Prudius and Andradóttir (2004, 2005a). For more detailed overviews on the topic, including discussion of simulation optimization techniques other than random search, the reader is referred to Andradóttir (1998, 2006), Fu (2002), and references therein.

In this paper we present two new variants of the SA algorithm. Both of the proposed methods use a decreasing cooling schedule as the means for controlling the probability of moving to seemingly inferior points and use the state with the highest estimated objective function value obtained from all the previous observations of the objective function values as the estimate of the optimal solution. This approach for estimating the optimal solution has been suggested by Andradóttir (1999). The only difference between our two variants of the SA algorithm is the way in which we estimate the objective function values at the current and candidate solutions in each iteration. In the first method, only observations obtained in the current iteration are used, while the second method utilizes all observations obtained so far at these two points. We also discuss under what conditions the proposed algorithms converge almost surely to the set of global optimal solutions  $\Theta^*$  and discuss the implications of our results for simulation optimization. Finally, we provide some numerical results which document the performance of our methods. For a more comprehensive development of our approaches, convergence proofs, and additional numerical results, the interested reader is referred to Prudius and Andradóttir (2005b).

The methods presented in this paper resemble the methods proposed and analyzed by Gelfand and Mitter (1989), Fox and Heine (1995), and Gutjahr and Pflug (1996) in that they are SA algorithms with a decreasing cooling schedule. In terms of the estimator of the optimal solution, our methods resemble Algorithm 2 in

Alrefaei and Andradóttir (1999) in that they use the state with the highest estimated objective function value as the estimator of the optimal solution. In our methods, we do not require the number of observations collected at the current and candidate solutions considered in a particular iteration to increase at a specific rate as the number of iterations grows, as is required in the method of Gelfand and Mitter (1989) and Gutjahr and Pflug (1996). Hence, our methods require less computation time per iteration as the number of iterations becomes large. In one of our variants of the SA algorithm, the number of observations generated per iteration is kept constant throughout the search (similar to Algorithm 2 of Alrefaei and Andradóttir 1999) and in the second variant, it can be adaptive (similar to the work of Fox and Heine 1995). Also in our methods, we allow the probability distribution that controls how a candidate solution is generated in the neighborhood of a current solution to depend on the iteration number.

Our second variant of the SA algorithm (see Section 3) is very similar to the SA algorithm of Fox and Heine (1995), with the only difference being the estimator of the optimal solution and hence the mode of convergence. In particular, both approaches use all observations obtained so far of the objective function value at each solution to estimate that objective function value. The convergence analysis presented by Fox and Heine (1995) shows that the sequence of current solutions generated by their variant of the SA algorithm converges in probability to the set  $\Theta^*$  provided that each feasible solution is sampled infinitely often with probability one and the SA algorithm for deterministic optimization converges in probability to the set  $\Theta^*$ . Rather than assuming that the sequence of current iterates samples each solution infinitely often with probability one, we provide conditions under which this happens. In fact, this is one of the main contributions in Prudius and Andradóttir (2005b).

This paper is organized as follows. In Sections 2 and 3 we present our new variants of the SA algorithm, discuss the conditions under which they converge to the set of global optimal solutions  $\Theta^*$ , and also discuss the implications of these results for simulation optimization. In Section 4 we provide some numerical results. Concluding remarks are given in Section 5.

## 2 A SIMULATED ANNEALING ALGORITHM

In this section, we present one of the SA algorithms proposed in Prudius and Andradóttir (2005b) and state its convergence properties. This algorithm utilizes a decreasing cooling schedule and the number of observations of the objective function values taken at the current and candidate solutions in each iteration is equal to a constant parameter  $K$ . As the estimator of the optimal solution, we use the state that has the highest estimated objective function value. For each  $n \in \mathbb{N}$  and  $\theta, \theta' \in \Theta$ , let  $Q_n(\theta, \theta')$  be the probability

of generating the candidate solution  $\theta'$  in iteration  $n$  when the current solution is  $\theta$ . We need the following definitions and assumptions:

**Assumption 1**  $\Theta$  is a finite element set.

**Assumption 2** We assume that  $Q_n \rightarrow Q$  as  $n \rightarrow \infty$ , where  $Q$  is an  $|\Theta| \times |\Theta|$  transition matrix of an irreducible Markov chain (MC) and  $|A|$  is the cardinality of any set  $A$ .

**Assumption 3** The cooling schedule  $\{T_n\}$  satisfies

$$T_{n+1} \leq T_n \text{ for all } n \in \mathbb{N},$$

$$\lim_{n \rightarrow \infty} T_n = 0.$$

Now we state a variant of the SA algorithm. Note that for all  $\theta \in \Theta$  and  $n \in \mathbb{N}$ ,  $C_n(\theta)$  is the number of times a solution  $\theta$  has been simulated in the first  $n$  iterations, and  $\Sigma_n(\theta)$  is the cumulative sum of all observations collected at the solution  $\theta$  in the first  $n$  iterations. For each  $n \in \mathbb{N}$ ,  $\theta_n$  is the current solution,  $\theta'_n$  is the candidate solution, and  $\theta_n^*$  is the estimator of the optimal solution in iteration  $n$ . Finally,  $[x]^+ = \max\{x, 0\}$  for all  $x \in \mathbb{R}$ .

### Algorithm 1

**Parameters:**  $\{Q_n\}$ ,  $\{T_n\}$ ,  $K$ .

**Step 0:** Choose a starting point  $\theta_0 \in \Theta$ . For all  $\theta \in \Theta$ , let  $\Sigma_0(\theta) = 0$  and  $C_0(\theta) = 0$ . Let  $n = 0$  and  $\theta_n^* = \theta_0$ .

**Step 1:** Given  $\theta_n = \eta$ , generate  $\theta'_n \in \Theta$  such that  $\mathbb{P}[\theta'_n = \eta' | \theta_n = \eta] = Q_n(\eta, \eta')$  for all  $\eta' \in \Theta$ .

**Step 2:** Given  $\theta_n = \eta$  and  $\theta'_n = \eta'$ , generate independent, identically distributed, and unbiased observations  $\{X_\eta^i\}_{i=1}^K$  of  $X_\eta$  and  $\{X_{\eta'}^i\}_{i=1}^K$  of  $X_{\eta'}$ , independent of any past observations. For  $\theta = \eta, \eta'$ , let

$$\Sigma_{n+1}(\theta) = \Sigma_n(\theta) + \sum_{i=1}^K h_\theta(X_\theta^i)$$

and  $C_{n+1}(\theta) = C_n(\theta) + K$ . Moreover, let  $C_{n+1}(\theta) = C_n(\theta)$  and  $\Sigma_{n+1}(\theta) = \Sigma_n(\theta)$  for all  $\theta \in \Theta$ ,  $\theta \neq \eta, \eta'$ . Calculate  $\hat{f}_n(\theta) = \Sigma_{n+1}(\theta)/K$  for  $\theta = \eta, \eta'$ .

**Step 3:** Given  $\theta_n = \eta$  and  $\theta'_n = \eta'$ , generate  $U_n \sim U[0, 1]$  (independently of all other random elements) and set

$$\theta_{n+1} = \begin{cases} \theta'_n & \text{if } U_n \leq G_n(\eta, \eta'), \\ \theta_n & \text{otherwise,} \end{cases}$$

where

$$G_n(\eta, \eta') = \exp \left[ \frac{-[\hat{f}_n(\eta) - \hat{f}_n(\eta')]^+}{T_n} \right].$$

**Step 4:** Let  $n = n + 1$  and select  $\theta_n^* \in \arg \max_{\theta \in \Theta} \Sigma_n(\theta)/C_n(\theta)$  (use the convention  $0/0 = -\infty$ ).

Go to Step 1.

Before stating our main convergence result for Algorithm 1, we need to give a few more definitions. For each  $\theta \in \Theta$ , let  $N(\theta) = \{\theta' \in \Theta : Q(\theta, \theta') > 0\}$ , which can be interpreted as the set of limiting neighbors of  $\theta$ . Let  $\tilde{\Theta}$  be the set of points that are local minima for the objective function  $f$  with respect to the neighborhood graph  $G$  induced by  $N$ , i.e.,

$$\tilde{\Theta} = \{\theta \in \Theta : f(\theta) \leq f(\theta'), \forall \theta' \in N(\theta)\}.$$

Observe that under Assumption 2, the condition  $\tilde{\Theta} = \Theta$  implies that  $f$  is constant on the feasible space  $\Theta$ . Hence, without loss of generality, we can assume that  $\tilde{\Theta}$  is a proper subset of  $\Theta$  because the optimization problem in (1) is trivial otherwise.

For each  $\theta \in \Theta$ , let  $\hat{f}(\theta) = \sum_{i=1}^K h_\theta(X_\theta^i)/K$ , where  $\{X_\theta^i\}_{i=1}^K$  are independent and identically distributed random elements with the distribution of  $X_\theta$ . Also assume that  $\{\hat{f}(\theta)\}_{\theta \in \Theta}$  are independent. Then define the maximum relative depth of the objective function in the neighborhood graph  $G$  as

$$L = \max_{\theta \in \Theta} \max_{\theta' \in N(\theta)} \mathbb{E}[\hat{f}(\theta) - \hat{f}(\theta')]^+.$$

Let

$$r = \max_{\theta \in \Theta} \max_{\theta' \in \Theta} d(\theta, \theta'),$$

where  $d(\theta, \theta')$  is the distance of  $\theta'$  from  $\theta$  measured by the length (number of edges) of the minimum length path from  $\theta$  to  $\theta'$  in  $G$  subject to the condition that the path contains at least one point in  $\Theta \setminus \tilde{\Theta} \neq \emptyset$ . Note that if  $r'$  is defined as  $r$  with the exception that the minimum length path need not contain a point in  $\Theta \setminus \tilde{\Theta}$  (so that  $r'$  can be viewed as a true diameter of the graph  $G$ ), then  $r \leq 2r'$ .

The following theorem has been proved by Prudius and Andradóttir (2005b).

**Theorem 1** Suppose that Assumptions 1, 2, and 3 hold and that the cooling schedule satisfies

$$\sum_{n=0}^{\infty} \exp\left(-\frac{rL}{T_n}\right) = +\infty. \quad (2)$$

Then the sequence  $\{\theta_n^*\}$  generated by Algorithm 1 converges almost surely to the set  $\Theta^*$ .

Also Prudius and Andradóttir (2005b) show that the Markov chain  $\{\theta_n\}$  generated by Algorithm 1 visits each feasible solution infinitely often with probability one, provided that Assumptions 1 through 3 and equation (2) hold and that  $\tilde{\Theta}$  is a proper subset of  $\Theta$ . This result is interesting from the following perspective. Suppose that the SA algo-

rithm for deterministic optimization studied by Hajek (1988) and Tsitsiklis (1989) (the same algorithm as Algorithm 1 with  $K = 1$ ,  $X_\theta$  being a degenerate random element, and  $h_\theta(X_\theta) = f(\theta)$  for all  $\theta \in \Theta$ ) converges in probability to the set  $\Theta^*$  (for conditions under which this happens see Hajek 1988 and Tsitsiklis 1989). These two results together imply that as  $n$  gets large, the sequence  $\{\theta_n\}$  tends to spend more time at “good” solutions, but still it visits every solution infinitely often.

### 3 A SIMULATED ANNEALING ALGORITHM WITH AVERAGING

In this section, we present a modification of Algorithm 1, state its convergence properties, and discuss the importance of this result. The main differences between Algorithms 1 and 2 are the following. If  $\theta$  is a candidate or a current solution, then the number of observations collected at  $\theta$  in iteration  $n$  is  $K_n(\theta)$ , which can depend on all the information gathered by the algorithm in the first  $n - 1$  iterations. Moreover, the estimate of the objective function value  $\hat{f}_n(\theta)$  is the average of all observations collected at  $\theta$  so far. These modifications allow us to weaken the assumption on the observations of the objective function values from being independent, identically distributed, and unbiased to just being strongly consistent, and still maintain the convergence guarantee of Algorithm 1 (under the additional assumptions given in Assumptions 5 and 6). We are now ready to state the following algorithm:

#### Algorithm 2

**Parameters:**  $\{Q_n\}$ ,  $\{T_n\}$ ,  $\{K_n\}$ .

**Step 0:** Identical to Step 0 of Algorithm 1.

**Step 1:** Identical to Step 1 of Algorithm 1.

**Step 2:** Given  $\theta_n = \eta$  and  $\theta'_n = \eta'$ , generate additional observations  $\{X_\eta^i\}_{i=1}^{K_n(\eta)}$  of  $X_\eta$  and  $\{X_{\eta'}^i\}_{i=1}^{K_n(\eta')}$  of  $X_{\eta'}$ . For  $\theta = \eta, \eta'$ , let

$$\Sigma_{n+1}(\theta) = \Sigma_n(\theta) + \sum_{i=1}^{K_n(\theta)} h_\theta(X_\theta^i)$$

and  $C_{n+1}(\theta) = C_n(\theta) + K_n(\theta)$ . Moreover, let  $C_{n+1}(\theta) = C_n(\theta)$  and  $\Sigma_{n+1}(\theta) = \Sigma_n(\theta)$  for all  $\theta \in \Theta$ ,  $\theta \neq \eta, \eta'$ . Calculate  $\hat{f}_n(\theta) = \Sigma_{n+1}(\theta)/C_{n+1}(\theta)$  for  $\theta = \eta, \eta'$ .

**Step 3:** Identical to Step 3 of Algorithm 1.

**Step 4:** Identical to Step 4 of Algorithm 1.

We need the following additional assumptions:

**Assumption 4** For all  $n \in \mathbb{N}$  and  $\theta \in \Theta$ ,  $K_n(\theta)$  is strictly positive whenever  $C_n(\theta) = 0$ . Moreover, for each  $\theta \in \Theta$ ,  $C_n(\theta) \rightarrow \infty$  as  $n \rightarrow \infty$  provided that  $\theta$  is sampled infinitely often.

**Assumption 5** For all  $n \in \mathbb{N}$  and  $\theta \in \Theta$ ,  $\hat{f}_n(\theta)$  can take only finitely many values; i.e., we assume finite precision in the knowledge of  $\hat{f}_n(\theta)$ .

**Assumption 6** For each  $\theta \in \Theta$ ,  $\hat{f}_n(\theta)$  is a strongly consistent estimator of  $f(\theta)$  provided that  $\theta$  is sampled infinitely often.

Note that Assumption 4 is satisfied if  $K_n(\theta) = K \in \mathbb{N}^+$  for all  $n \in \mathbb{N}$  and  $\theta \in \Theta$ . Also, we do not consider Assumption 5 to be a serious restriction because most practical applications of the SA algorithm are implemented on computers which have finite precision. The same assumption is used in Fox and Heine (1995).

Let  $L = \max_{\theta \in \Theta} \max_{\theta' \in N(\theta)} [f(\theta) - f(\theta')]^+$ . The following theorem has been proved by Prudius and Andradóttir (2005b).

**Theorem 2** Suppose that Assumptions 1 through 6 hold and that the cooling schedule satisfies equation (2). Then the sequence  $\{\theta_n^*\}$  generated by Algorithm 2 converges almost surely to the set  $\Theta^*$ .

Prudius and Andradóttir (2005b) also show that the sequence  $\{\theta_n\}$  generated by Algorithm 2 converges to the set  $\Theta^*$  in probability provided that the conditions in Theorem 2 are satisfied and Algorithm 2 for deterministic optimization converges in probability to the set  $\Theta^*$  (for conditions under which this is true, see Hajek 1988 and Tsitsiklis 1989). The main difference between this result and the result of Fox and Heine (1995) is that we do not assume that Algorithm 2 samples each solution infinitely often with probability one, but rather show that this happens under some mild conditions.

We next discuss the implications of Theorem 2 for simulation optimization. Note that  $K_n(\theta)$  is allowed to depend on the information gathered by the algorithm by iteration  $n - 1$ . This is an extension of the previous work, where  $K_n(\theta)$  is usually controlled deterministically. This extension might be useful, for instance, in situations where a lot of observations have been collected at  $\theta$  by iteration  $n - 1$  so that the variance of  $\hat{f}_n(\theta)$  is small, and hence collecting more observations at this point  $\theta$  might not produce a considerably better estimate of  $f(\theta)$ . In this situation, it might be desirable to keep  $K_n(\theta)$  small. The identification of good strategies for choosing the parameters  $K_n(\theta)$  is beyond the scope of this work.

Another important implication of Theorem 2 is the following. Algorithm 2 is convergent almost surely to  $\Theta^*$  even with strongly consistent estimators of  $f(\theta)$ ; i.e., as long as  $\sum_{i=1}^n h_\theta(X_\theta^i)/n \rightarrow f(\theta)$  almost surely as  $n \rightarrow \infty$  for all  $\theta \in \Theta$ . This is a particularly useful result for steady-state simulation. In particular, Prudius and Andradóttir (2005b) show that Algorithm 2 converges almost surely to  $\Theta^*$  when  $f$  is a steady-state performance measure. For a more detailed discussion on this, the interested reader is referred to Prudius and Andradóttir (2005b).

## 4 A NUMERICAL EXAMPLE

In this section, we present numerical results obtained by applying Algorithms 1 and 2 to solve the discrete simulation optimization problem (1) with  $\Theta = \{\theta = (\theta_1, \theta_2) \in \mathbb{N}^2 : 0 \leq \theta_1, \theta_2 \leq 49\}$ ,  $f(\theta) = \max\{f_1(\theta), f_2(\theta), 0\}$ , and  $h_\theta(X_\theta) = f(\theta) + X_\theta$ , where  $f_1(\theta) = -(0.4\theta_1 - 5)^2 - 2(0.4\theta_2 - 17.2)^2 + 7$ ,  $f_2(\theta) = -(0.4\theta_1 - 12)^2 - (0.4\theta_2 - 4)^2 + 4$ , and  $X_\theta$  is a normally distributed random variable with mean 0 and variance 50 for each  $\theta \in \Theta$ . This objective function is of interest because it has two hills of different heights (4 and 6.96), located relatively far apart (the hill of height 4 is centered at (30, 10) and the hill of height 6.96 is centered at (12, 43) and (13, 43)), and separated by a flat valley (of height 0). Note that the standard deviation of the noise is roughly equal to the range of the objective function values. This makes the response surface highly noisy and hence this problem is relatively difficult to solve. This numerical problem was also used in the numerical studies of Prudius and Andradóttir (2004, 2005a).

Now we describe the implementation details. We apply Algorithms 1 and 2 using two different neighborhood structures. The first neighborhood structure is given by

$$N_0(\theta) = \{(\theta'_1, \theta'_2) \in \Theta \setminus \{\theta\} : |\theta'_i - \theta_i| \leq 1 \text{ for } i = 1, 2\}.$$

For this neighborhood structure, for all  $\theta \in \Theta$  and  $n \in \mathbb{N}$ , we take  $Q_n(\theta, \theta') = 1/|N_0(\theta)|$  for all  $\theta' \in N_0(\theta)$  and 0 otherwise. This implementation of Algorithm 1(2) will be referred to as the Local Algorithm 1(2). The second neighborhood structure is given by  $N_1(\theta) = \Theta \setminus \{\theta\}$  for all  $\theta \in \Theta$ . For this neighborhood structure, for all  $\theta \in \Theta$  and  $n \in \mathbb{N}$ , we take  $Q_n(\theta, \theta') = 1/|N_1(\theta)|$  for all  $\theta' \in N_1(\theta)$  and  $Q_n(\theta, \theta) = 0$ . This implementation of Algorithm 1(2) will be referred to as the Global Algorithm 1(2).

The parameters for Algorithms 1 and 2 are selected based on Theorems 1 and 2 above to guarantee the almost sure convergence of each method. In particular, the cooling schedule  $\{T_n\}$  for each optimization method is of the form  $T_n = C/\log(n + 10)$  for all  $n \in \mathbb{N}$ , where  $C$  is a positive constant. The value of  $C$  is 565, 20, 312, and 14 for Local Algorithm 1, Global Algorithm 1, Local Algorithm 2, and Global Algorithm 2, respectively (these values are reasonably tight upper bounds on the product  $rL$ , see equation (2)). The number  $K$  of objective function observations collected at the current and candidate solutions is 10 for Local and Global Algorithms 1, while the number  $K_n(\theta_n)$  and  $K_n(\theta'_n)$  of objective function observations collected at the current and candidate solutions in iteration  $n$  is equal to 2 for Local and Global Algorithms 2, independent of the past information. We choose a relatively large value for  $K$  because of the high variance of  $X_\theta$  and smaller values for  $K_n(\theta)$ , where  $n \in \mathbb{N}$  and  $\theta \in \Theta$ , because the averaging in Algorithm 2 reduces the impact of this large variance. The

initial solution is selected randomly for all four algorithms. The performance of the algorithms is compared based on 100 independent replications. We used common random numbers in our experiment in the sense that the initial seeds for the uniforms required for choosing the initial state in Step 0 of the algorithms, generating the candidate solutions in Step 1, estimating the objective function values in Step 2, and selecting  $\theta_{n+1}$  in Step 3 are the same. Figure 1 shows the average performance of the four approaches as the simulation effort increases.

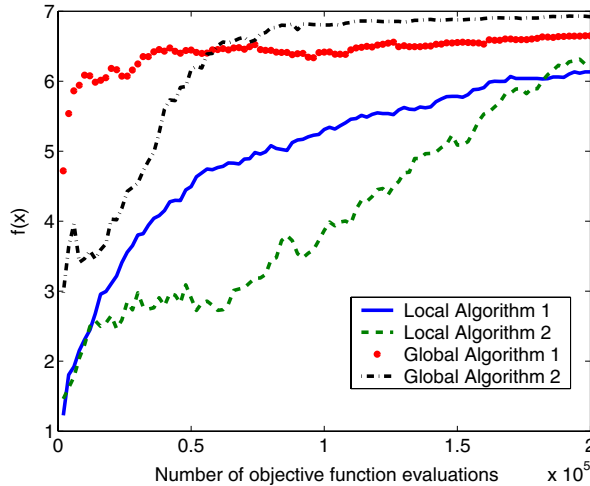


Figure 1: Average Objective Function Value at the Estimated Optimal Solution Versus the Number of Objective Function Evaluations

It is obvious from Figure 1 that Global Algorithms 1 and 2 perform considerably better in this example than their local counterparts. This can be explained by the fact that an initial solution might be far away from the subregions containing good solutions and it might take the two local algorithms many iterations to identify a good subregion. The reason why Global Algorithm 2 performs worse than Global Algorithm 1 in the beginning of the search is that Global Algorithm 2 collects only 2 observations of the objective function values at the current and candidate solutions (as opposed to 10 in Global Algorithm 1) and hence, the objective function estimates are initially very noisy. But as the search progresses, Global Algorithm 2 becomes more conservative in terms of moving to worse points because better objective function estimates become available and the temperature values are much lower (note that the  $C$  value for Global Algorithm 2 is lower than the  $C$  value for Global Algorithm 1 and that Global Algorithm 2 completes more iterations than Global Algorithm 1 for a given amount of computer effort because  $K$  is greater than  $K_n(\theta)$  for all  $\theta \in \Theta$ ). The above discussion suggests that Global Algorithm 2 might perform better if  $K_n(\theta)$  were decreasing in  $n$  for each  $\theta$ ; see also Section 3.

## 5 CONCLUSION

In this paper, we have proposed two new variants of the SA algorithm with a decreasing cooling schedule for solving discrete simulation optimization problems. Our methods differ from each other only in the way in which the estimates of the objective function values at the current and candidate solutions are obtained in each iteration. In the first method, only observations obtained in the current iteration are used, while the latter method utilizes all observations obtained so far at these points. Our approaches are guaranteed to converge almost surely to the set of global optimal solutions under mild conditions. A numerical example has been provided that documents the performance of our approaches. More numerical studies are of course required to adequately understand the behavior of the proposed methods and to compare our approaches to competing methods.

## ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 0000135, Grant No. 0217860, and Grant No. 0400260.

## REFERENCES

- Alrefaei, M. H., and S. Andradóttir. 1999. A simulated annealing algorithm with constant temperature for discrete stochastic optimization. *Management Science* 45 (5): 748–764.
- Alrefaei, M. H., and S. Andradóttir. 2001. A modification of the stochastic ruler method for discrete stochastic optimization. *European Journal of Operational Research* 133 (1): 160–182.
- Alrefaei, M. H., and S. Andradóttir. 2005. Discrete stochastic optimization using variants of the stochastic ruler method. *Naval Research Logistics* 52 (4): 344–360.
- Andradóttir, S. 1995. A method for discrete stochastic optimization. *Management Science* 41 (12): 1946–1961.
- Andradóttir, S. 1996. A global search method for discrete stochastic optimization. *SIAM Journal on Optimization* 6 (2): 513–530.
- Andradóttir, S. 1998. Simulation optimization. In *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, ed. J. Banks, 307–333. New York: Wiley.
- Andradóttir, S. 1999. Accelerating the convergence of random search methods for discrete stochastic optimization. *ACM Transactions on Modeling and Computer Simulation* 9 (4): 349–380.
- Andradóttir, S. 2006. An overview of simulation optimization via random search. To appear in the *Handbooks in Operations Research and Management Science: Simu-*

- lation, ed. S. G. Henderson and B. L. Nelson. Elsevier Science.
- Fox, B. L., and G. W. Heine. 1995. Probabilistic search with overrides. *The Annals of Applied Probability* 5 (4): 1087–1094.
- Fu, M. C. 2002. Optimization for simulation: Theory vs. practice. *INFORMS Journal on Computing* 14 (3): 192–215.
- Gelfand, S. B., and S. K. Mitter. 1989. Simulated annealing with noisy or imprecise energy measurements. *Journal of Optimization Theory and Applications* 62 (1): 49–62.
- Gong, W.-B., Y.-C. Ho, and W. Zhai. 1999. Stochastic comparison algorithm for discrete optimization with estimation. *SIAM Journal on Optimization* 10 (2): 384–404.
- Gutjahr, W. J., and G. C. Pflug. 1996. Simulated annealing for noisy cost functions. *Journal of Global Optimization* 8 (1): 1–13.
- Hajek, B. 1988. Cooling schedules for optimal annealing. *Mathematics of Operations Research* 13 (2): 311–329.
- Hong, L. J., and B. L. Nelson. 2005. Discrete optimization via simulation using COMPASS. *Operations Research*, to appear.
- Pichtlanken, J., and B. L. Nelson. 2003. A combined procedure for optimization via simulation. *ACM Transactions on Modeling and Computer Simulation* 13 (2): 155–179.
- Prudius, A. A., and S. Andradóttir. 2004. Simulation optimization using balanced explorative and exploitative search. In *Proceedings of the 2004 Winter Simulation Conference*, ed. R. G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, 545–549. Institute of Electrical and Electronics Engineers: Piscataway, New Jersey.
- Prudius, A. A., and S. Andradóttir. 2005a. Balanced explorative and exploitative search with estimation for simulation optimization. Working paper.
- Prudius, A. A., and S. Andradóttir. 2005b. Stochastic optimization using variants of the simulated annealing algorithm. Working paper.
- Shi, L., and S. Ólafsson. 2000. Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability* 2 (3): 271–291.
- Tsitsiklis, J. N. 1989. Markov chains with rare transitions and simulated annealing. *Mathematics of Operations Research* 14 (1): 70–90.
- Yan, D., and H. Mukai. 1992. Stochastic discrete optimization. *SIAM Journal on Control and Optimization* 30 (3): 594–612.

## AUTHOR BIOGRAPHIES

**ANDREIA. PRUDIUS** is a Ph.D. candidate in the School of Industrial and Systems Engineering at the Georgia Institute

of Technology. He received a B.S. in Industrial Engineering from Boğaziçi University in 2001 and an M.S. in Operations Research from the Georgia Institute of Technology in 2004. His research interests are in simulation, and especially in simulation optimization. His e-mail address is <aprudius@isye.gatech.edu>.

**SIGRÚN ANDRADÓTTIR** is a Professor in the School of Industrial and Systems Engineering at the Georgia Institute of Technology. She received a B.S. in Mathematics from the University of Iceland in 1986, an M.S. in Statistics from Stanford University in 1989, and a Ph.D. in Operations Research from Stanford University in 1990. Her research interests include stochastic optimization, simulation, and stochastic processes. Her e-mail address is <sa@isye.gatech.edu>.