Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

SIMULATION-BASED RETROSPECTIVE OPTIMIZATION OF STOCHASTIC SYSTEMS: A FAMILY OF ALGORITHMS

Jihong Jin

1929 Danube Way
Bolingbrook, IL 60490, U.S.A.

Bruce Schmeiser

School of Industrial Engineering
Purdue University
West Lafayette, IN 47907, U.S.A.

ABSTRACT

We consider optimizing a stochastic system, given only a simulation model that is parameterized by continuous decision variables. The model is assumed to produce unbiased point estimates of the system performance measure(s), which must be expected values. The performance measures may appear in the objective function and/or in the constraints. We develop a family of retrospective-optimization (RO) algorithms based on a sequence of sample-path approximations to the original problem with increasing sample sizes. Each approximation problem is obtained by substituting point estimators for each performance measure and using common random numbers over all values of the decision variables. We assume that these approximation problems can be deterministically solved to within a specified error in the decision variables, and that this error is decreasing to zero. The computational efficiency of RO arises from being able to solve the next approximation problem efficiently based on knowledge gained from the earlier, easier approximation problems.

1 INTRODUCTION

In many practical situations one is required to optimize a stochastic system where some functions of the system are not available analytically, but can be estimated via Monte Carlo simulation. We focus on a retrospective approach to such problems, and apply it to the following class of problems.

$$
\begin{array}{llll}
 & \text{minimize} & f_0(\mathbf{x}) & = \mathbb{E}\{y_0(\mathbf{x},\boldsymbol{\omega})\}, \\
(\mathrm{P}_0) & \text{subject to} & f_j(\mathbf{x}) & = \mathbb{E}\{y_j(\mathbf{x},\boldsymbol{\omega})\} \\
 & & & = 0, \qquad j = 1, ..., p < d, \\
 & & f_j(\mathbf{x}) & = \mathbb{E}\{y_j(\mathbf{x},\boldsymbol{\omega})\} \\
 & & & \geq 0, \qquad j = p+1, ..., p+q, \\
 & & & \mathbf{x} \in X \subset \mathbb{R}^d,
\end{array}
$$

where $y_j(\mathbf{x},\boldsymbol{\omega})$, $j = 0, ..., p+q$, are the scalar sample performances driven by a stochastic effect $\boldsymbol{\omega} \in \Omega$ defined with respect to the underlying probability space $(\Omega, \mathcal{F}, P)$. The decision-variable vector $\mathbf{x}$ is selected from a variable set X, a closed compact subset of $\mathbb{R}^d$. The functions $f_j(\mathbf{x})$, $j = 0, ..., p+q$, are the expectations of $y_j(\mathbf{x},\boldsymbol{\omega})$, $j = 0, ..., p+q$, with respect to the probability measure P, respectively. We assume that $p < d$; $p = d$, no optimization problem exists in general and the optimum $\mathbf{x}^*$ is determined directly from the simultaneous solution of the system of equations $f_j(\mathbf{x}) = \mathbb{E}\{y_j(\mathbf{x},\boldsymbol{\omega})\} = 0$, $j = 1, ..., d$. The optimum $\mathbf{x}^*$ is assumed to exist and to be unique.

In many realistic applications, and especially when the stochastic effect $\boldsymbol{\omega}$ has a large dimensionality, it is typically impossible to calculate the expected values for $f_j(\mathbf{x})$, $j = 0, ..., p+q$, in closed form, or to obtain them via quadrature. Hence, numerical approximations are required to estimate the expected-value functions, and problem P_0 is solved by using only the estimators of $f_j(\mathbf{x})$, $j = 0, ..., p+q$. This paper is to develop methodologies and algorithms to solve this class of stochastic optimization problems.

The remainder of this paper is organized as follows. In Section 2 we introduce a simulation-based sample-path approximation approach to estimate problem P_0. Next we propose a family of retrospective optimization algorithms based on Monte Carlo simulation techniques to solve problem P_0 in Section 3. We then discuss general guidelines for the algorithm implementation in Section 4. A concluding remark along with a future research area is given in Section 5.

2 THE SIMULATION-BASED SAMPLE-PATH APPROXIMATION APPROACH

This section is concerned with estimating of the expectations of sample performances $y_j(\mathbf{x},\boldsymbol{\omega})$, $j = 0, ..., p+q$, using simulation techniques. After discussing a simulation-based sample-path approximation approach to estimate problem

P$_0$, a simple example is given to help the reader understand the discussion.

2.1 The Sample-Path Approximation of Problem P$_0$

The simulation-based sample-path approximation approach to problem P$_0$ is based on Monte Carlo simulation techniques. The basic idea is that common random effects $\boldsymbol{\zeta}$ are used to estimate $f(\mathbf{x})$ for all $\mathbf{x} \in X$. More specifically, let $y(\mathbf{x}, \boldsymbol{\omega}_1), ..., y(\mathbf{x}, \boldsymbol{\omega}_m)$ be a series of random samples obtained by simulating m (usually independent) stochastic effects $\boldsymbol{\zeta} = \{\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, ..., \boldsymbol{\omega}_m\}$ with respect to the underlying probability space $(\Omega, \mathcal{F}, P)$, and consequently the expected-value function $f(\mathbf{x}) = \mathbb{E}\{y(\mathbf{x}, \boldsymbol{\omega})\}$ is approximated by the sample average function

$$\bar{y}(\mathbf{x}, \boldsymbol{\zeta}) = \frac{\sum_{t=1}^{m} y(\mathbf{x}, \boldsymbol{\omega}_t)}{m}. \tag{1}$$

Furthermore, the sample-path approximation problem P corresponding to problem P$_0$ is

$$\begin{array}{lll} & \text{minimize} & \bar{y}_0(\mathbf{x}, \boldsymbol{\zeta}), \\ (\text{P}) & \text{subject to} & \bar{y}_j(\mathbf{x}, \boldsymbol{\zeta}) = 0, \quad j = 1, ..., p < d, \\ & & \bar{y}_j(\mathbf{x}, \boldsymbol{\zeta}) \geq 0, \quad j = p+1, ..., p+q, \\ & & \mathbf{x} \in X \subset I\!R^d. \end{array}$$

The sample-path approximations $\bar{y}_j(\mathbf{x}, \boldsymbol{\zeta})$, $j = 0, ..., p+q$, are functions of the decision-parameter vector $\mathbf{x}$ and the stochastic effects $\boldsymbol{\zeta}$. We let $\mathbf{X}^*(\boldsymbol{\zeta})$ denote the sample-path optimum of problem P. The sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta})$ is random, depending upon the random effects $\boldsymbol{\zeta}$.

2.2 A Simple Bus-Scheduling Example

In a bus station, passengers arrive and wait for buses. For the sake of simplicity, we assume infinite bus capacity such that when a bus arrives, it takes all the passengers who are waiting in the station. Figure 1 depicts the passenger arrival process. Time starts at zero. The station is empty. (There would be a bus that left at time zero and took away all passengers). We let T_k, $k = 1, 2, \ldots$, denote the kth passenger arrival time, A_k, $k = 1, 2, \ldots$, denote the time between the arrivals of the $(k-1)$th passenger and the kth passenger, where A_1 is the time until the first passenger arrival. The distribution functions F_{A_k}, $k = 1, 2, \ldots$, of interarrival times are known. Originally, there is only one bus, which leaves at time T. To better serve the customers, the second bus is added to the schedule and leaves at time x $(0 \leq x \leq T)$. The goal is to determine when the second bus should leave the station so as to minimize the expected total wait time of all passengers in time $[0, T]$.

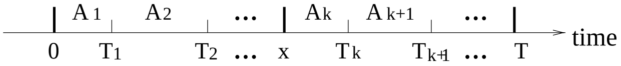

Figure 1: Passenger Arrival Process

A mathematical formulation corresponding to the problem is

$$\begin{array}{ll} \text{minimize} & \mathbb{E}\{y_0(x, \boldsymbol{\omega})\}, \\ \text{subject to} & x \in [0, T], \end{array} \tag{2}$$

where $y_0(x, \boldsymbol{\omega})$ is a sample performance measure of total wait time of all passengers in time [0, T] driven by a stochastic effect $\boldsymbol{\omega}$. Let $W_k(x, \boldsymbol{\omega})$ denote the kth passenger wait time. Then $y_0(x, \boldsymbol{\omega})$ can be written as

$$y_0(x, \boldsymbol{\omega}) = \sum_{k=1}^{\infty} W_k(x, \boldsymbol{\omega}) I(T_k(\boldsymbol{\omega}) \leq T). \tag{3}$$

The indicator function $I(T_k(\boldsymbol{\omega}) \leq T) = 1$ for $T_k(\boldsymbol{\omega}) \leq T$ and $I(T_k(\boldsymbol{\omega}) \leq T) = 0$ for $T_k(\boldsymbol{\omega}) > T$. It implies that the total wait time $y_0(x, \boldsymbol{\omega})$ counts only those passengers who arrive before time T. The kth passenger wait time $W_k(x, \boldsymbol{\omega})$ is defined as

$$W_k(x, \boldsymbol{\omega}) = \left\{ \begin{array}{ll} x - T_k(\boldsymbol{\omega}) & \text{if } T_k(\boldsymbol{\omega}) \leq x \\ T - T_k(\boldsymbol{\omega}) & \text{if } T_k(\boldsymbol{\omega}) > x \end{array} \right. . \tag{4}$$

Furthermore, it is easy to see from Figure 1 that the kth passenger arrival time $T_k(\boldsymbol{\omega})$ is

$$T_k(\boldsymbol{\omega}) = \sum_{j=1}^{k} A_j(\boldsymbol{\omega}), \tag{5}$$

where the interarrival times $A_k(\boldsymbol{\omega})$, $k = 1, 2, \ldots$, have the distribution functions F_{A_k}, $k = 1, 2, \ldots$, respectively, which are given.

This is a one-dimensional stochastic optimization problem with one explicit deterministic constraint. Because (in the general case) the objective function $\mathbb{E}\{y_0(x, \boldsymbol{\omega})\}$ is not closed form, an estimate of problem 2 is used. In the rest of this section we illustrate the simulation-based sample-path approximation approach to estimate Problem 2.

In the simulation-based sample-path approximation approach to problem 2, each stochastic effect $\boldsymbol{\omega}$ is a sequence of random-number seeds used for simulating a passenger arrival process, and sample performance $y_0(\mathbf{x}, \boldsymbol{\omega})$ depends on the passenger arrival process driven by the sequence of random-number seeds $\boldsymbol{\omega}$. Following is a summary of the procedure to compute a sample performance $y_0(\mathbf{x}, \boldsymbol{\omega})$.

1. Generate a sequence of uniform random variables $u(\omega_1), u(\omega_2), \ldots$ by a sequence of random number seeds $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots)$.

2. Calculate $A_k = F_{A_K}^{-1}(u(\omega_k))$ for $k = 1, 2, \ldots$, where $F_{A_K}^{-1}$ is the inverse function of F_{A_K}. Note that F_{A_K}'s are known, therefore are the $F_{A_K}^{-1}$'s.
3. Calculate $T_k(\boldsymbol{\omega}) = \sum_{j=1}^{k} A_j(\boldsymbol{\omega})$ for $k = 1, 2, \ldots$ until $T_k(\boldsymbol{\omega}) > T$.
4. Calculate $W_k(x, \boldsymbol{\omega})$, for $k = 1, 2, \ldots$, by Equation (4).
5. Calculate $y_0(\mathbf{x}, \boldsymbol{\omega})$ by Equation (3).

Repeat the above simulation experiment using m independent random-number seed sequences $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_m$, and obtain m sample performances $y_0(\mathbf{x}, \boldsymbol{\omega}_1), \ldots, y_0(\mathbf{x}, \boldsymbol{\omega}_m)$. Then a sample-path approximation of Problem 2 is

$$\begin{array}{ll} \text{minimize} & \bar{y}_0(x, \boldsymbol{\zeta}) = \sum_{t=1}^{m} y_0(x, \boldsymbol{\omega}_t)/m, \\ \text{subject to} & x \in X = [0, T]. \end{array} \qquad (6)$$

3 A FAMILY OF RETROSPECTIVE OPTIMIZATION ALGORITHMS

A retrospective technique is different from a prospective technique, in which a fixed set of feasible values of $\mathbf{x}$ is explored to look for the solution, much like a prospector searches for gold: a spot is chosen for exploration because good results are foreseen. A retrospective technique, on the other hand, determines the solution after all uncertainties have been known. In the real world it is rare for a retrospective technique to have the luxury of hindsight or the task of simply predicting the past. We do however have the benefit of hindsight in a simulated system performance. The basic philosophy of the retrospective approach to optimize simulated system is due to Schruben (1991). The notion is to observe the output from a simulation retrospectively to generate optimal solutions as if the outcomes of all uncertainties were known in advance. The retrospective concept has previously been employed independently in Rubinstein (1991) and Healy (1992).

In this paper, we explore the retrospective philosophy to problem P_0, and use the sample-path approximation described in Section 2 to estimate problem P_0. The sample-path approximation problem P is obtained by simulating the stochastic effects $\boldsymbol{\zeta}$. Once the $\boldsymbol{\omega}_t$'s are fixed, solving problem P becomes a deterministic optimization problem. A realization of the optimum $\mathbf{X}^*(\boldsymbol{\zeta})$ is obtained by retrospectively solving a deterministic optimization problem with respect to the sample-path approximation problem P as if the outcomes of all uncertainties have been known. Note that the sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta})$ may not exist or may not be unique for a finite sample size.

A natural estimate of the optimum $\mathbf{x}^*$ of problem P_0 is the average of n independent realizations of the optimum $\mathbf{X}^*(\boldsymbol{\zeta})$ obtained from independently seeded replications of problem P. To obtain n independent realizations of $\mathbf{X}^*(\boldsymbol{\zeta})$, this method requires retrospectively to solve n independently seeded replications of problem P, and during each deterministic search for a realization of $\mathbf{X}^*(\boldsymbol{\zeta})$, many $\mathbf{x}$'s may be explored and each $\mathbf{x}$ must be evaluated m times. Therefore the computational effort can be substantial, especially when the starting point is not accurate.

Chen (1994) proposes a retrospective approximation (RA) algorithm to solve a one-dimensional stochastic root finding problem. RA iteratively solves a sequence of deterministic sample-path equations for a sequence of retrospective roots, with increasing sample sizes. A final estimate of the true root is computed from those solutions. Shapiro (1996) discusses the analogous idea for continuous-variable optimization. This analogy is the basis for our family of algorithms.

Instead of finding independent realizations of the sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta})$ from independently seeded replications of a single sample-path approximation problem P, we extend the general idea of RA to problem P_0 and propose a family of retrospective optimization (RO) algorithms. RO iteratively solves a sequence of sample-path approximation problems P_i, $i = 1, 2, \ldots$,

$$(\mathrm{P}_i) \quad \begin{array}{lll} \text{minimize} & \bar{y}_0(\mathbf{x}, \boldsymbol{\zeta}_i), & \\ \text{subject to} & \bar{y}_j(\mathbf{x}, \boldsymbol{\zeta}_i) = 0, & j = 1, ..., p < d, \\ & \bar{y}_j(\mathbf{x}, \boldsymbol{\zeta}_i) \geq 0, & j = p+1, ..., p+q, \\ & \mathbf{x} \in X \subset I\!\!R^d, & \end{array}$$

where $\bar{y}_j(\mathbf{x}, \boldsymbol{\zeta}_i) = \sum_{t=1}^{m_i} y_j(\mathbf{x}, \boldsymbol{\omega}_t)/m_i$, $j = 0, 1, 2, ..., p + q$, and $\boldsymbol{\zeta}_i = \{\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_{m_i}\}$ is generated independently for each i. The sample-size sequence m_i, $i = 1, 2, ...$, strictly increases. At each iteration, sample-path approximation problem P_i is retrospectively solved within an error tolerance ϵ_i. The error tolerance ϵ_i, $i = 1, 2, ...$, is a positive sequence, decreasing to zero. Let $\mathbf{X}^*(\boldsymbol{\zeta}_i)$ be the ith *sample-path optimum* of problem P_i, $\mathbf{X}(\boldsymbol{\zeta}_i)$ the ith *retrospective solution* obtained by deterministically solving sample-path problem P_i within ϵ_i such that $\|\mathbf{X}(\boldsymbol{\zeta}_i) - \mathbf{X}^*(\boldsymbol{\zeta}_i)\| < \epsilon_i$, where $\| \cdot \|$ denotes the distance between $\mathbf{X}(\boldsymbol{\zeta}_i)$ and $\mathbf{X}^*(\boldsymbol{\zeta}_i)$, and i the *retrospective iteration number*. After the ith retrospective iteration, $\mathbf{x}^*$ is approximated by the ith *retrospective estimate* $\bar{\mathbf{X}}_i$, a function of $\mathbf{X}(\boldsymbol{\zeta}_1), ..., \mathbf{X}(\boldsymbol{\zeta}_i)$. More specifically, RO works as follows.

3.1 A Family of RO Algorithms

1. Initialize the retrospective iteration number $i = 1$, the sample size m_1, the error tolerance ϵ_1 and the initial point $\bar{\mathbf{X}}_0$.
2. Generate $\boldsymbol{\zeta}_i = (\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_{m_i})$ independently of $\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_{i-1}$.
3. Deterministically solve sample-path approximation problem P_i, in which the information obtained in previous $(i-1)$ retrospective iterations such as

$\sqrt{\widehat{\mathbb{E}}\{\|\bar{\mathbf{X}}_{i-1} - \mathbf{X}^*(\boldsymbol{\zeta}_i)\|^2\}}$ and $\bar{\mathbf{X}}_{i-1}$ are used in the ith deterministic search. Return any one solution $\mathbf{X}(\boldsymbol{\zeta}_i)$ satisfying $\|\mathbf{X}(\boldsymbol{\zeta}_i) - \mathbf{X}^*(\boldsymbol{\zeta}_i)\| < \epsilon_i$.

4. Compute $\bar{\mathbf{X}}_i$ from $\mathbf{X}(\boldsymbol{\zeta}_1), \ldots, \mathbf{X}(\boldsymbol{\zeta}_i)$.
5. Stop? If yes, return $\bar{\mathbf{X}}_i$. Otherwise, compute m_{i+1} and ϵ_{i+1}, let $i \leftarrow i+1$, and go to step 2.

The decreasing tolerance ϵ_i and increasing sample size m_i are used to ensure that the algorithms converge. When the sample-size sequence m_i, $i = 1, 2, ...$, strictly increases, the sample-path approximation problem P_i converges to problem P_0, and the sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta}_i)$ converges to $\mathbf{x}^*$. When m_i is large, the sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta}_i)$ is a better estimate of the optimum $\mathbf{x}^*$ because the sample-path approximation problem P_i mimics problem P_0 well. Furthermore, if the error tolerance ϵ_i is small, the retrospective solution $\mathbf{X}(\boldsymbol{\zeta}_i)$ is close to $\mathbf{X}^*(\boldsymbol{\zeta}_i)$, and hence close to $\mathbf{x}^*$.

4 GENERAL GUIDELINES FOR THE IMPLEMENTATION

In implementing the algorithms five decisions that affect their efficiency must be made:

- Method for deterministically solving the sample-path approximation problem P_i,
- Rule for initiating m_1, ϵ_1 and $\bar{\mathbf{X}}_0$,
- Rule for computing error tolerance ϵ_i, for $i = 2, 3, \ldots$,
- Rule for successively increasing m_i, for $i = 2, 3, \ldots$,
- Rule for computing the ith retrospective estimate $\bar{\mathbf{X}}_i$ from $\mathbf{X}(\boldsymbol{\zeta}_1), \ldots, \mathbf{X}(\boldsymbol{\zeta}_i)$, for $i = 1, 2, \ldots$,
- Rule for stopping the whole procedure.

First, the error tolerance ϵ_i should progressively decrease as the sample size m_i increases such that sample-path approximation problem P_i is loosely solved at the early retrospective iterations, and more tightly solved only as problem P_i converges to problem P_0. In addition, as i increases, though more tightly solving problem P_i enlarges the computational effort, the knowledge gained from solving previous retrospective iterations could help in reducing the computational effort. For example, the $(i-1)$th retrospective estimate $\bar{\mathbf{X}}_{i-1}$ and $\sqrt{\widehat{\mathbb{E}}\{\|\bar{\mathbf{X}}_{i-1} - \mathbf{X}^*(\boldsymbol{\zeta}_i)\|^2\}}$ have provided a better starting point and information about possible distance between starting point $\bar{\mathbf{X}}_{i-1}$ and the ith sample-path optimum $\mathbf{X}^*(\boldsymbol{\zeta}_i)$ in the ith deterministic search. Therefore, the overall computational effort required in the procedure is reduced.

Second, the sample-size sequence m_i, $i = 1, 2, ...$, and the stopping rule should be considered together. The rate of increasing m_i has a tradeoff: quick increase makes the sample-path approximation problems converge to problem P_0 fast, but also enlarges the computational effort fast. The stopping rule of the whole procedure, i.e., the rule for determining which problem P_i to stop at, compromises between simulation run length and the quality (accuracy and precision) of the retrospective estimate $\bar{\mathbf{X}}_i$.

Third, the rule for computing the ith retrospective estimate $\bar{\mathbf{X}}_i$ from all previous retrospective solutions $\mathbf{X}(\boldsymbol{\zeta}_1), \ldots, \mathbf{X}(\boldsymbol{\zeta}_i)$ should depend on the sample sizes $m_1, \ldots, m_i$, since the retrospective solution $\mathbf{X}(\boldsymbol{\zeta}_i)$ comes closer and closer to $\mathbf{x}^*$ as the sample size m_i increases and the error tolerance ϵ_i decreases. A typical function form would be a sample-size weighted average of $\mathbf{X}(\boldsymbol{\zeta}_1), \ldots, \mathbf{X}(\boldsymbol{\zeta}_i)$, that is

$$\bar{\mathbf{X}}_i = \frac{\sum_{h=1}^{i} m_h \mathbf{X}(\boldsymbol{\zeta}_h)}{\sum_{h=1}^{i} m_h}. \tag{7}$$

Fourth, the method for deterministically solving problem P_i should be efficient. One of choices is to use the flexible-tolerance method described in Himmelblau (1972) to implement the deterministic search because it is very efficient. The flexible-tolerance method improves the value of the objective function by using information provided by feasible points, as well as certain nonfeasible points termed near-feasible points. Many constrained programming methods, on the other hand, spend a considerable portion of the computational effort to satisfy rather rigorous feasibility requirements. This is inefficient, especially when the constraints of problem P_i do not mimic those of problem P_0 well with small sample size m_i. In the flexible-tolerance method, the near-feasibility limit is progressively made more restrictive as the deterministic search proceeds toward the optimum $\mathbf{X}^*(\boldsymbol{\zeta}_i)$, until in the limit a retrospective solution $\mathbf{X}(\boldsymbol{\zeta}_i)$ is found satisfying $\|\mathbf{X}(\boldsymbol{\zeta}_i) - \mathbf{X}^*(\boldsymbol{\zeta}_i)\| < \epsilon_i$. Furthermore, if the error tolerance ϵ_i goes to zero, the near-feasibility limit can be made even more restrictive until in the limit only feasible points are accepted. Hence, the constraints are loosely satisfied when the sample size m_i is small, and more tightly satisfied only as the constraints mimic those in problem P_0 well.

5 CONCLUSIONS

In this article, we introduce a family of RO algorithms to optimize a stochastic system where only a simulation model is given. RO are based on a sequence of sample-path approximations to the original problem. Each approximation problem is obtained by substituting point estimators for each performance measure and using common random numbers over all values of the decision variables. The sequence of approximation problems is obtained by increasing the number of simulation replications. We assume that these approximation problems, which are deterministic, can be solved to within a specified error in the decision variables, and that this error is decreasing to zero. The computational efficiently of RO arises from being able to solve the next approximation problem efficiently based on knowledge gained from the earlier, easier approximation problem.

The discussion of the convergence and the implementation guidelines in this article is the intuitive outlines to deal with RO. Further research on the statistical properties of estimates derived from these algorithms, and a proof of the convergence will be presented in a separated paper.

ACKNOWLEDGMENTS

This work was supported in part by Purdue Research Foundation under Grant No. 690-1287-2501.

REFERENCES

Chen, H. 1994. *Stochastic Root Finding in System Design*. PhD thesis, Purdue University.

Healy, K. 1992. *Optimizing Stochastic Systems: A Retrospective/Deterministic Approach*. PhD thesis, Cornell University.

Himmelblau, D. 1972. *Applied Nonlinear Programming*. McGraw-Hill, Inc.

Rubinstein, R. 1991. How to optimize discrete-event systems from a single sample path by the score function method. *Ann. Oper. Res.*, Vol. 27:175-212.

Schruben, L. 1991. Path-wise Decomposition and retrospective optimization of some simulation experiments. Technical Report NPSOR-9119, Naval Postgraduate School.

Serfling, R. 1980. *Approximation Theorems of Mathematical Statistics*. John Wiley & Sons, New York.

Shapiro, A. 1996. Simulation based optimization. In *Proc. Winter Simulation Conf.*, pages 332-336.

AUTHOR BIOGRAPHIES

JIHONG JIN received a Ph.D from the School of Industrial Engineering at Purdue University. She is a member of INFORMS and ASQ.

BRUCE SCHMEISER is a professor in the School of Industrial Engineering at Purdue University. His interests lie in applied operations research, with emphasis in stochastic models, especially the probabilistic and statistical aspects of stochastic simulation. Since 1976, he has been an active participant in the Winter Simulation Conference, including being Program Chair in 1983 and chairing the Board of Directors during 1988–1990.