Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

ON CHOOSING PARAMETERS IN RETROSPECTIVE-APPROXIMATION ALGORITHMS FOR SIMULATION-OPTIMIZATION

Raghu Pasupathy

Industrial and Systems Engineering
Virginia Tech
Blacksburg, VA 24061, U.S.A.

ABSTRACT

The Simulation-Optimization (SO) problem is a constrained optimization problem where the objective function is observed with error, usually through an oracle such as a simulation. Retrospective Approximation (RA) is a general technique that can be used to solve SO problems. In RA, the solution to the SO problem is approached using solutions to a sequence of approximate problems, each of which is generated using a specified sample size and solved to a specified error tolerance. In this paper, our focus is *parameter* choice in RA algorithms, where the term *parameter* is broadly interpreted. Specifically, we present (i) conditions that guarantee convergence of estimated solutions to the true solution; (ii) convergence properties of the sample-size and error-tolerance sequences that ensure that the sequence of estimated solutions converge to the true solution in an optimal fashion; and (iii) a numerical procedure that efficiently solves the generated approximate problems for one-dimensional SO.

1 INTRODUCTION AND PROBLEM STATEMENT

The Deterministic-Optimization (DO) problem is that of finding a minimizer x^* of a real-valued function $G : \mathfrak{D} \subset \Re^q \to \Re$. This problem has been extensively studied in the last thirty years, and efficient algorithms for solving DO problems exist at least in contexts where the function G satisfies certain structural assumptions (Bazaara et al. 2006, Bertsekas 1999).

The SO problem is a generalization of the DO problem where the objective is the same, i.e., a minimizer x^* of G is sought, but the function G is known only through a consistent estimator. The consistent estimator of G is usually constructed by averaging responses from an oracle such as a Monte Carlo simulation. Formally, the SO problem is stated as follows.

Given: An oracle capable of generating, for any $x \in \mathfrak{D} \subset \mathbf{R}^q$, an estimator $\overline{Y}_m(x)$ of the function $G(x) : \mathfrak{D} \to \mathbf{R}$ such that $\overline{Y}_m(x) \to G$ uniformly with probability one (w.p.1) as $m \to \infty$.

Find: A local minimizer $x^* \in \mathfrak{D}$ of G, i.e., find x^* having a neighborhood $V(x^*)$ such that every $x \in V(x^*)$ satisfies $G(x) \geq G(x^*)$, assuming that one such x^* exists.

As stated, the SO problem makes no assumptions about the nature of $\overline{Y}_m(x)$ except that $\lim_{m\to\infty} \overline{Y}_m(x) = G(x)$ uniformly w.p.1. Also, the feasible set $\mathfrak{D}$ is assumed to be known in the sense that the functions involved in the specification of $\mathfrak{D}$ are observed without error. Various slightly differing flavors of the SO problem have appeared in the literature. See, for example, Nemirovski and Shapiro (2004), Ruszczynski and Shapiro (2003), and Fu (1994).

RA is a general technique that can be used to solve problems such as SO, and is based on the following simple idea. Since the actual problem contains functions that cannot be observed exactly, approach the solution to the actual problem using solutions to a sequence of approximate problems, each of which is generated using a specified finite sample size and solved to a specified error tolerance. So, during the kth iteration, a *sample-path problem* is generated using a sample size m_k, and solved to within error tolerance ε_k. Across iterations, the sample size m_k is gradually increased to ∞ and the error tolerance ε_k is gradually decreased to 0.

In order to implement RA algorithms on a problem, various *parameters* in the RA framework need to be specified by the user. Here, the term *parameters* is broadly used, and includes the sequences $\{m_k\}$ and $\{\varepsilon_k\}$, and a numerical procedure that can be used to solve the generated sample-path problem to within the specified error tolerance. Our main focus in this paper is providing guidance on choosing parameters in RA algorithms for solving SO problems. We try to answer three questions that arise in this context.

(i) Under what conditions do RA algorithms for SO problems converge?

(ii) How should the parameter sequences $\{m_k\}$ and $\{\varepsilon_k\}$ be chosen in RA algorithms?

(iii) What numerical procedures can be advantageously employed for solving the sample-path problem?

The answer to the first question, discussed in Section 3, is an almost trivial consequence of various existing results that have been derived recently in the context of sample average approximation. We address the second question in Section 4, and the third question in Section 5, although only partially. For brevity, most results are stated without proofs.

2 LITERATURE REVIEW

In this section, we present an overview of the important works related to solving SO problems. We limit our discussion to works that are gradient-based, i.e., methods that rely on estimated measures of the gradient of the objective function, and seeking local extrema. See Fu (1994) and Safizadeh (1990) for a commentary on SO problems in general.

2.1 Stochastic Approximation (SA)

Much of the existing literature on gradient-based methods are variants of Classical Stochastic Approximation (CSA), the original root-finding algorithm proposed by Robbins and Monro (1951). CSA, as proposed by Robbins and Monro, has the simple iterative structure

$$X_{k+1} = X_k - a_k \overline{Y}_k(X_k), \tag{1}$$

where k is the iteration number, X_0 is an initial guess of the root x^* of the root-finding equation $g(x) = 0$, $\overline{Y}_k$ is an unbiased estimator of the root-finding function $g(x)$, and $\{a_k\}_{k=0}^{\infty}$ is a predetermined sequence of positive constants satisfying $\sum_{k=0}^{\infty} a_k = \infty$ and $\sum_{k=0}^{\infty} a_k^2 < \infty$. CSA's iterates converge to x^* satisfying $g(x^*) = 0$ in mean square under fairly general conditions. There has since been much work done in accelerating the convergence of CSA. We do not go into details here since this is in the context of root finding. See for instance Kesten (1958), Venter (1967), Wasan (1969), Andradóttir (1990) and Andradóttir (1991).

CSA is relevant to SO problems if the root-finding function g is interpreted as the gradient of an objective function G whose minimizer we seek in SO problems. This is the basis of the Kiefer-Wolfowitz procedure (Kiefer and Wolfowitz 1952), which is apparently the first known algorithm for solving one-dimensional SO problems. In the Kiefer-Wolfowitz procedure, the gradient of the observable function G is approximated using the method of finite differences, and an iteration similar to that of CSA produces a sequence of iterates that converge to the minimizer of G in probability. Blum (1954) extends the Kiefer-Wolfowitz procedure to multiple dimensions and proves almost sure convergence. Since Blum's work in 1954, there have been several variants of the Kiefer-Wolfowitz procedure. See Fabian (1968), Kushner and Clark (1978), and more recently Simultaneous Perturbation Stochastic Approximation (SPSA) in Spall (1998) and Spall (2000). Most of these works have focused on accelerating convergence of the Kiefer-Wolfowitz procedure through improved gradient estimation methods. He, Fu, and Marcus (2003) extend the convergence proofs presented by Spall to the case when G is convex and not necessarily differentiable.

2.2 Sample Average Approximation (SAA)/ Retrospective Approximation (RA)

SAA, also known by various other names including sample-path optimization, stochastic counterpart method and retrospective approximation, is another general technique used for solving SO problems. It appears that the first reference to this technique is by Healy and Schruben (1991). Several other authors have used the technique in various contexts. See for instance Rubinstein and Shapiro (1993), Plambeck et al. (1996), and Atlason, Epelman, and Henderson (2002). The idea of SAA is easily stated. Instead of solving the actual SO problem, solve an approximate problem S obtained by substituting the unknown underlying objective function G by the sample-path approximation $\overline{y}_m(x;\underline{\omega})$. The sample-path approximation $\overline{y}_m(x;\underline{\omega})$ is the realization of the consistent estimator $\overline{Y}_m(x)$ of $G(x)$, generated using the vector of random numbers $\underline{\omega} = \{\omega_1, \omega_2, \ldots, \omega_m\}$ and sample size m. So, the sample-path problem S has the following simple form:

$$\begin{array}{ll} \text{Minimize} & \overline{y}_m(x;\underline{\omega}) \quad (S) \\ \text{subject to} & x \in \mathfrak{D} \subset \mathbf{R}^q. \end{array}$$

So in SAA, instead of the original SO problem, a *single* approximate deterministic problem S generated with a "sufficiently large" sample size is solved to optimality using an appropriately chosen non-linear programming algorithm. Under certain conditions, most notably on $\overline{y}_m(x;\underline{\omega})$, the solution to S converges to a minimizer x^* of G as the sample size $m \to \infty$. See Shapiro and de Mello (2000), Shapiro (2000), Ruszczynski and Shapiro (2003), and de Mello (2003) for a thorough discussion of SAA including results on the rates of convergence.

RA, proposed by Chen and Schmeiser (2001) in the context of stochastic root finding, is a variant of SAA where, instead of generating and solving a *single* sample-path problem S, a *sequence* of sample-path problems $\{S_k\}$ are generated with increasing sample sizes $\{m_k\} \to \infty$, and solved to decreasing tolerances $\{\varepsilon_k\} \to 0$. The philosophy behind the RA structure is as follows: during the early iterations, i.e., for small k, use small sample sizes m_k

and large error tolerances ε_k in solving the sample-path problem S_k; in later iterations, as the estimated solution $\overline{X}_k$ tends closer to a minimizer x^*, use larger sample sizes and correspondingly smaller error tolerances. The early iterations are efficient because the small sample sizes ensure that not much computing effort is expended in generating the sample-path problem, and the later iterations are efficient because the estimated solution $\overline{X}_k$ is probably close to a minimizer x^* and not much effort is expended in solving the sample-path problem S_k. This structure of RA is central to the simultaneous goals of proving convergence and achieving good practical performance. Specifically, the RA framework is as follows.

Parameters: (i) Numerical procedure for solving the sample-path problem S_k to within a specified error tolerance ε_k for each k; (ii) An initial sample size m_1 and a rule for successively increasing m_k for $k \geq 2$; (iii) A rule for computing an error-tolerance sequence $\{\varepsilon_k\}$ that goes to zero w.p.1.

Logic:

0. Initialize the retrospective iteration number $k = 1$. Set m_1 and ε_1.
1. Generate the vector of random numbers $\underline{\omega}_k$.
2. Use the numerical method to solve the deterministic sample-path problem S_k to within error tolerance ε_k and obtain a retrospective solution X_k.
3. Compute estimated solution $\overline{X}_k$ as a weighted combination of the retrospective solutions $\{X_j\}_{j=1}^k$.
4. If the stopping criterion is not satisfied, compute m_{k+1} and ε_{k+1}. Set $k \leftarrow k+1$ and go to 1.

It is worth noting again that RA is only a broad framework and specific algorithms result only upon choosing parameters (i), (ii), and (iii) in the framework.

3 CONDITIONS FOR GUARANTEED CONVERGENCE

In this section, we present two simple results that establish sufficient conditions to ensure that the sequence $\{X_k\}$ of retrospective solutions in RA algorithms converges to the true solution x^* w.p.1. We start with Theorem 1 which is a well-known result that appears in various forms in a number of papers including Shapiro (2000) and Dupačová and Wets (1988). Theorem 1 asserts that the set of local minimizers converges (in distance) to the unique minimizer of the limiting function G if the feasible set $\mathfrak{D}$ is compact, the function G is continuous, and the sequence of sample-path functions $\overline{y}_{m_k}(x;\underline{\omega}_k)$ converges to G uniformly w.p.1.

Theorem 1 *Assume that*

(i) *the set $\mathfrak{D} \subset \boldsymbol{R}^q$ is compact;*
(ii) *the function $G : \mathfrak{D} \to \boldsymbol{R}$ is continuous with a unique minimizer $x^* \in \mathfrak{D}$;*
(iii) *the function $\overline{y}_{m_k}(x;\underline{\omega}_k)$ is such that the sequence $\{\overline{y}_{m_k}(x;\underline{\omega}_k)\}$ converges to G* uniformly *w.p.1, i.e., $\sup\{|\overline{y}_{m_k}(x;\underline{\omega}_k) - G(x)|\} \to 0$ as $k \to \infty$ w.p.1.*

If $L_k \subset \mathfrak{D}$ is the set of local minimizers of $\overline{y}_{m_k}(x;\underline{\omega}_k)$, and if L_k is non-empty for large enough k, then the distance $d(L_k, \{x^*\}) \to 0$ as $k \to \infty$ w.p.1.

In the above result, the assumption that the set of local minimizers L_k is non-empty holds if, for instance, $\overline{y}_{m_k}(x;\underline{\omega}_k)$ is continuous. Without assumptions such as the continuity of $\overline{y}_{m_k}(x;\underline{\omega}_k)$, nothing can be said about the limiting behavior of L_k even if $\overline{y}_{m_k}(x;\underline{\omega}_k)$ converges uniformly to G, and G has a unique minimizer. As an example, let $G(x) = x^2$ for $x \in [-1,1]$, and let $\overline{y}_{m_k}(x;\underline{\omega}_k) = x^2$ for $x \in [-1,0)$, and $\overline{y}_{m_k}(x;\underline{\omega}_k) = x^2 + 1/k$ for $x \in [0,1]$. For this example, $\overline{y}_{m_k}(x;\underline{\omega}_k)$ converges to G uniformly, and L_k is empty for all k, even though G has a unique minimizer at $x^* = 0$.

Recall that in RA algorithms the kth sample-path problem is solved to within the chosen tolerance ε_k, i.e., the kth retrospective solution X_k is ε_k within some local minimizer X_k^* of the sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$ w.p.1. A trivial consequence of this fact and Theorem 1 is that the sequence $\{X_k\}$ of retrospective solutions converges to the true solution x^* w.p.1. Theorem 2 states this formally.

Theorem 2 *Assume that the conditions in Theorem 1 hold. Furthermore, assume that*

(i) *the positive-valued sequence $\{\varepsilon_k\} \to 0$ w.p.1;*
(ii) *for each k, X_k satisfies $\|X_k - X_k^*\| \leq \varepsilon_k$ w.p.1, where X_k^* is some local minimizer of $\overline{y}_{m_k}(x;\underline{\omega}_k)$.*

Then $\{X_k\} \to x^*$ w.p.1.

Proof: Since X_k satisfies $\|X_k - X_k^*\| \leq \varepsilon_k$ w.p.1, the random variable $\Delta_k = X_k^* - X_k$ is supported on a hypersphere of radius ε_k centered at the origin w.p.1. So, $\{\Delta_k\} \to 0$ w.p.1. However, $X_k = X_k^* + \Delta_k$, and from Theorem 1, $\{X_k^*\} \to x^*$ w.p.1. Conclude that $\{X_k\} \to x^*$ w.p.1. □

4 CHOOSING THE ERROR-TOLERANCE SEQUENCE

Recall that in RA algorithms, $\{m_k\}$ is the chosen sequence of sample sizes used to generate the sample-path problems across iterations, ε_k is the chosen tolerance to which the kth sample-path problem is solved, $\{X_k\}$ is the sequence of retrospective solutions obtained across iterations, $\{X_k^*\}$ is a corresponding sequence of local minimizers satisfying $\|X_k - X_k^*\| \leq \varepsilon_k$, and x^* is the true solution. It is worth noting here that there may be many possible sequences $\{X_k^*\}$ of local minimizers satisfying $\|X_k - X_k^*\| \leq \varepsilon_k$. In order to avoid this complication, we assume, for now, that $\overline{y}_{m_k}(x;\underline{\omega}_k)$ has a unique minimum.

Our objective, in this section, is providing insight into choosing a sequence of error tolerances $\{\varepsilon_k\}$ that ensures

rapid convergence of X_k to x^*. As a measure of efficiency, we use the mean squared error (MSE) of X_k computed as

$$\mathrm{MSE}(X_k,x^*) = \mathrm{E}[(X_k-x^*)^T(X_k-x^*)],$$

where all vectors involved are column vectors. Also, for ease of exposition, we decompose the MSE of X_k^* as

$$\mathrm{MSE}(X_k^*,x^*) = \mathrm{E}[(X_k^*-x^*)^T(X_k^*-x^*)] = v_k + b_k^2,$$

where $v_k = \mathrm{E}[\left(X_k^*-\mathrm{E}[X_k^*]\right)^T\left(X_k^*-\mathrm{E}[X_k^*]\right)]$ and $b_k^2 = \left(\mathrm{E}[X_k^*]-x^*\right)^T\left(\mathrm{E}[X_k^*]-x^*\right)$. Notice that v_k is the trace (sum of diagonal elements) of the covariance matrix of X_k^*.

In Theorem 3, we establish the *minimum rate* at which the sequence of tolerances $\{\varepsilon_k\}$ should converge to zero to ensure the optimal convergence of $\mathrm{MSE}(X_k,x^*)$. Specifically, Theorem 3 states that it is best to choose the sequence of error tolerances $\{\varepsilon_k\}$ so that it converges to zero at least as fast as the sequence $\{\mathrm{Max}(v_k,b_k^2)\}$.

Theorem 3 *Assume that*

- *(i)* *the set $\mathfrak{D} \subset \mathbb{R}^q$ is compact;*
- *(ii)* *the function $G:\mathfrak{D}\to\mathbb{R}$ is continuous and convex with a unique minimizer $x^*\in\mathfrak{D}$;*
- *(iii)* *the functional sequence $\{\overline{y}_{m_k}(x;\underline{\omega}_k)\}$ converges to G uniformly w.p.1, and each function $\overline{y}_{m_k}(x;\underline{\omega}_k)$ has a unique minimum;*
- *(iv)* *the positive-valued sequence of tolerances $\{\varepsilon_k\}\to 0$ w.p.1;*

Then

- *(a)* $\limsup_{k\to\infty}\varepsilon_k^2/\mathrm{Max}(v_k,b_k^2)=\infty$ *w.p.1 implies* $\limsup_{k\to\infty}\mathrm{MSE}(X_k,x^*)/\mathrm{Max}(v_k,b_k^2)=\infty$;
- *(b)* $\limsup_{k\to\infty}\varepsilon_k^2/\mathrm{Max}(v_k,b_k^2)<\infty$ *w.p.1 implies* $\limsup_{k\to\infty}\mathrm{MSE}(X_k,x^*)/\mathrm{Max}(v_k,b_k^2)<\infty$.

Theorem 3 establishes the *minimum rate* at which the sequence of tolerances $\{\varepsilon_k\}$ should converge to zero, i.e., it suggests that the sequence $\{\varepsilon_k\}$ should converge to zero at least as fast as the sequence $\{\mathrm{Max}(v_k,b_k^2)\}$.

The assumptions appearing in Theorem 3, especially that $\overline{y}_{m_k}(x;\underline{\omega}_k)$ has a unique minimum, can be relaxed. This would involve the careful specification of the sequence $\{X_k^*\}$. We do not go into the generalization details here since the result, as stated, captures the essential idea.

We now present Theorem 4 which is useful in deciding the *maximum rate* of convergence of the sequence of tolerances $\{\varepsilon_k\}$. Specifically, Theorem 4 asserts that as long as the sequence of error tolerances $\{\varepsilon_k^2\}$ converges to zero at least as fast as the sequence $\{\mathrm{Max}(v_k,b_k^2)\}$, irrespective of how much faster it converges, the convergence rate of $\{\mathrm{MSE}(X_k,x^*)\}$ remains the same. In other words, all error-tolerance sequences satisfying $\limsup_{k\to\infty}\varepsilon_k^2/\mathrm{Max}(v_k,b_k^2)<\infty$ w.p.1 are equivalent from the standpoint of asymptotic efficiency.

Theorem 4 *Let the sequence $\{\varepsilon_k\}$ satisfy $0<\limsup_{k\to\infty}\varepsilon_k^2/\left(\mathrm{Max}(v_k,b_k^2)\right)^{1+\nu}<\infty$ w.p.1 for some $\nu\geq 0$. Then*

$$\limsup_{k\to\infty}\frac{\mathrm{MSE}(X_k,x^*)}{\mathrm{Max}(v_k,b_k^2)} = \limsup_{k\to\infty}\frac{v_k+b_k^2}{\mathrm{Max}(v_k,b_k^2)}.$$

Theorem 4 suggests that there is no gain in the convergence rate of the sequence of mean squared errors $\{\mathrm{MSE}(X_k,x^*)\}$ with an increase in the convergence rate of the sequence of error tolerances $\{\varepsilon_k\}$, as long as $\{\varepsilon_k\}$ converges to zero at least as fast as the sequence $\{\mathrm{Max}(v_k,b_k^2)\}$.

Recall that the kth error tolerance ε_k is an indicator of how much effort is expended in solving the kth sample-path problem. Since solving to lower tolerances usually involves increased computing effort, it seems intuitively clear that choosing ε_k so that it converges to zero at the same rate as $\{\mathrm{Max}(v_k,b_k^2)\}$ will be "most efficient." Theorem 5 validates this intuition.

In Theorem 5, the measure "work $\times$ squared error," computed as $\mathrm{E}\left[N_k(X_k-x^*)^T(X_k-x^*)\right]$, is used in assessing algorithm performance. In the expression $\mathrm{E}\left[N_k(X_k-x^*)^T(X_k-x^*)\right]$, the random variable N_k represents the number of simulation calls expended during the kth iteration, in solving the sample-path problem to tolerance ε_k. So, smaller values of $\mathrm{E}\left[N_k(X_k-x^*)^T(X_k-x^*)\right]$ indicate better algorithm performance.

Theorem 5 *Assume that*

- *(i)* $0<\limsup_{k\to\infty}\varepsilon_k^2/\left(\mathrm{Max}(v_k,b_k^2)\right)^{1+\nu}<\infty$ *w.p.1 for some $\nu\geq 0$;*
- *(ii)* $\limsup_{k\to\infty}\{\varepsilon_k^2 E[N_k]\}<\infty$ *w.p.1; and*
- *(iii)* *N_k is independent of X_k.*

Then

- *(a)* $\nu>0$ *implies*

$$\limsup_{k\to\infty} E\left[N_k(X_k-x^*)^T(X_k-x^*)\right]=\infty;$$

- *(b)* $\nu=0$ *implies*

$$\limsup_{k\to\infty} E\left[N_k(X_k-x^*)^T(X_k-x^*)\right]<\infty.$$

The assumption (ii) in Theorem 5 is motivated by the frequent scenario where the squared error in the reported solution of a minimization problem is of the order of the reciprocal of the computing effort expended. Clearly, a result similar to Theorem 5 will hold even when the computing effort involved is greater. Also, while assumption (ii) can probably be relaxed without weakening Theorem 5, it is still less stringent

than assuming, for instance, that $\limsup_{k\to\infty}\{\varepsilon_k^2 N_k\}$ is uniformly bounded w.p.1.

Theorems 3, 4 and 5 together imply that, under certain conditions, choosing the sequence of error tolerances $\{\varepsilon_k\}$ so that it converges to zero at same rate as the sequence $\{\text{Max}(v_k, b_k^2)\}$ is *optimal*. Can we find one such sequence of error tolerances? This question is especially interesting since v_k and b_k^2 are usually unknown. Theorem 6 answers this question using the well-known result that, under certain conditions, $\sqrt{m_k}(X_k^* - x^*)$ has a non-degenerate limiting distribution. Results of this sort appear in, for instance, Shapiro (2000) and Dupačová and Wets (1988).

Theorem 6 *Assume that the conditions in Theorem 3 hold, and that $\overline{y}_{m_k}(x;\underline{\omega}_k)$ is the average of m_k independent and identically distributed (i.i.d) random variables. If the sequence of error tolerances $\{\varepsilon_k\}$ satisfies $0 < \limsup_{k\to\infty}\varepsilon_k\sqrt{m_k} < \infty$ w.p.1, then $0 < \limsup_{k\to\infty}\varepsilon_k^2/\text{Max}(v_k, b_k^2) < \infty$. w.p.1.*

For any chosen sequence of sample sizes $\{m_k\}$, Theorem 6 provides guidance on the choice of the sequence of error tolerances $\{\varepsilon_k\}$. For example, choosing $\varepsilon_k = c_3/\sqrt{m_k}$, for any constant $c_3 > 0$, is a simple way of ensuring that $0 < \limsup_{k\to\infty}\varepsilon_k^2/\text{Max}(v_k, b_k^2) < \infty$ w.p.1. A more sophisticated choice is inspired by assuming that the retrospective solutions $\{X_k\}$ are independently distributed random variables, and that X_k has variance σ^2/m_k, where $\sigma^2 > 0$ is some underlying covariance matrix to be estimated from observed data.

5 AN RA ALGORITHM FOR THE ONE-DIMENSIONAL CASE

In the preceding sections, we provided guidance on the choice of the sequence of error tolerances $\{\varepsilon_k\}$ in RA algorithms. We did not focus on the choice of the sequence of sample sizes $\{m_k\}$ because $\{\varepsilon_k\}$ is adjusted based on $\{m_k\}$. So, asymptotically, the only major effect that the sequence $\{m_k\}$ will probably have is in deciding the number of *stopping points*, i.e., on how often the algorithm reports the current solution to the user. For example, using $m_1 = 1$ and then increasing sample size by 10% in each iteration has worked in practice.

In this section, we focus on yet another parameter in RA algorithms, namely the numerical procedure that is used to satisfy the iteration-level termination criterion. Recall that, for purposes of this paper, the termination criterion for the kth iteration is identifying a retrospective solution X_k so that $\|X_k - X_k^*\| \le \varepsilon_k$, where X_k^* is some local minimizer of the sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$. In the following paragraphs, we describe an algorithm that achieves this efficiently for the one-dimensional SO case.

5.1 Overview

The broad idea of the proposed numerical procedure is as follows. During the kth iteration, we generate a sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$ with sample size m_k, and then seek three points X_a, X_b, X_c such that the following conditions hold:

C.1 $X_a < X_b < X_c$;
C.2 $X_c - X_a \le \varepsilon_k$;
C.3 $\overline{y}_{m_k}(X_a;\underline{\omega}_k) \ge \overline{y}_{m_k}(X_b;\underline{\omega}_k)$; and
C.4 $\overline{y}_{m_k}(X_c;\underline{\omega}_k) \ge \overline{y}_{m_k}(X_b;\underline{\omega}_k)$.

It can be shown that if points X_a, X_b, X_c satisfy C.1, C.2, C.3, and C.4, then there exists a local minimizer in the interval $[X_a, X_c]$. So, any point $X_k \in [X_a, X_b]$ satisfies the termination criterion. It is worth noting that, counter to what intuition may suggest, the conditions C.1, C.2, C.3, and C.4 do not guarantee the existence of a local minimizer of the sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$ in the absence of continuity.

A physical interpretation of C.1, C.2, C.3, and C.4 is that they jointly stipulate the identification of three points X_a, X_b, X_c within ε_k from each other such that a quadratic constrained to pass through the points $(X_a, \overline{y}_{m_k}(X_a;\underline{\omega}_k))$, $(X_b, \overline{y}_{m_k}(X_b;\underline{\omega}_k))$, and $(X_c, \overline{y}_{m_k}(X_c;\underline{\omega}_k))$ attains its minimum. This interpretation is also useful in multidimensional extension.

We illustrate successful termination of the kth iteration in Figure 1 where the dotted curve represents the graph of the unknown underlying function G, and the solid curve is the graph of the sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$. The points X_a, X_b, X_c in Figure 1 satisfy conditions C.1, C.2, C.3, and C.4. The kth retrospective solution X_k is then appropriately chosen from the interval $[X_a, X_c]$.

We next describe a method to identify points X_a, X_b, X_c satisfying conditions C.1, C.2, C.3, and C.4.

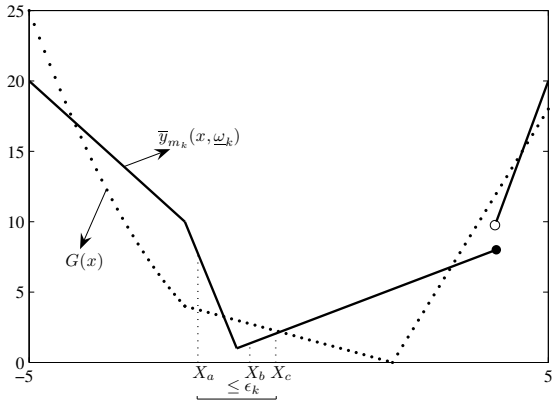

Figure 1: Illustration of the Successful Termination of the kth Iteration

5.2 Numerical Procedure Logic

The strategy is to choose a direction ($+x$ or negative $-x$) and simulate at points separated by increasing step sizes until we encounter three points that satisfy conditions C.1, C.3 and C.4. Figure 2 illustrates this idea. If the three identified points also satisfy condition C.2, the iteration terminates successfully. Otherwise, a technique such as bisection search is used to identify three other points that satisfy condition C.2 in addition to C.1, C.3, and C.4. The motivation behind this strategy is the idea that if the sample-path function approximates the underlying function G even "reasonably" well, which it will at least eventually as the sample size m_k increases, the procedure will succeed in identifying the required points when G has a unique minimum. The following paragraphs provide more detail on this procedure.

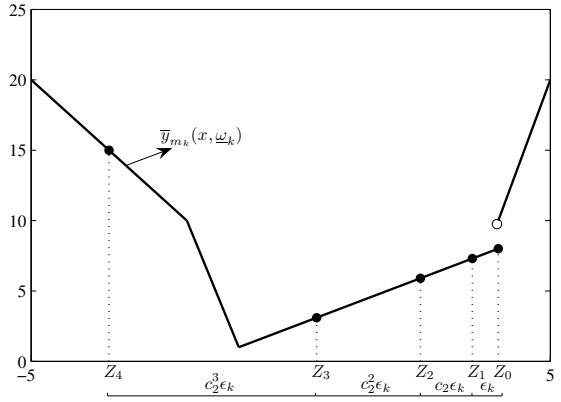

Figure 2: Illustration of Algorithm Logic During the kth Iteration

Recall that $\overline{X}_{k-1}$ is the estimated solution at the end of $k-1$ iterations. It forms the initial solution for the kth iteration. During the kth iteration, we first simulate at design points $\overline{X}_{k-1}$ and $\overline{X}_{k-1}+d\varepsilon_k$ to obtain responses $\overline{y}_{m_k}(\overline{X}_{k-1};\underline{\omega}_k)$ and $\overline{y}_{m_k}(\overline{X}_{k-1}+d\varepsilon_k;\underline{\omega}_k)$ respectively. Here, d is chosen to be 1 or -1, and $\varepsilon_k > 0$ is the chosen step size.

For ease of exposition, relabel as Z_0, that amongst $\overline{X}_{k-1}, \overline{X}_{k-1}+d\varepsilon_k$ which has the larger observed response. Likewise, relabel as Z_1, that amongst $\overline{X}_{k-1}, \overline{X}_{k-1}+d\varepsilon_k$ which has the smaller observed response. So, if $\overline{y}_{m_k}(\overline{X}_{k-1};\underline{\omega}_k) \geq \overline{y}_{m_k}(\overline{X}_{k-1}+d\varepsilon_k;\underline{\omega}_k)$ then

$$\begin{aligned} Z_0 &= \overline{X}_{k-1}, Z_1 = \overline{X}_{k-1}+d\varepsilon_k; \text{ and} \\ Z_0 &= \overline{X}_{k-1}+d\varepsilon_k, Z_1 = \overline{X}_{k-1} \text{ otherwise.} \end{aligned}$$

With this relabeling, we see that we have simulated at the design points Z_0 and Z_1, and that $\overline{y}_{m_k}(Z_0;\underline{\omega}_k) \geq \overline{y}_{m_k}(Z_1;\underline{\omega}_k)$.

Let $t = \text{sign}(Z_1 - Z_0)$ and define $Z_2 = Z_1 + tc_2\varepsilon_k, Z_3 = Z_2 + tc_2^2\varepsilon_k, \ldots$, where the step-size multiplier c_2 is some constant greater than 1. Recalling that we have already simulated at Z_0 and Z_1, simulate at the design points $Z_2, Z_3, \ldots$, until three points Z_{n-2}, Z_{n-1}, Z_n are encountered such that $\overline{y}_{m_k}(Z_{n-2};\underline{\omega}_k) \geq \overline{y}_{m_k}(Z_{n-1};\underline{\omega}_k)$, and $\overline{y}_{m_k}(Z_n;\underline{\omega}_k) \geq \overline{y}_{m_k}(Z_{n-1};\underline{\omega}_k)$. We illustrate this in Figure 2. The points Z_{n-2}, Z_{n-1}, Z_n satisfy all stipulations for successful termination except, perhaps, C.2. So, we repeatedly bisect $[Z_{n-2}, Z_n]$ to find three points X_a, X_b, X_c that satisfy C.1, C.2, C.3 and C.4.

We now present the pseudo-code for the non-terminating version of the proposed numerical procedure.

5.3 Procedure Listing

Given: Default values for the initial guess $\overline{X}_0$, step-size multiplier $c_2 > 1$, and a method to compute the sequences $\{\varepsilon_k\}$ and $\{m_k\}$.

Find: A local minimizer $x^* \in \mathfrak{D}$ of G.

0. Initialize $k = 0$.
1. Set $k = k+1$.
2. Simulate at $\overline{X}_{k-1}$ and $\overline{X}_{k-1}+\varepsilon_k$ to get $\overline{y}_{m_k}(\overline{X}_{k-1};\underline{\omega}_k)$ and $\overline{y}_{m_k}(\overline{X}_{k-1}+\varepsilon_k;\underline{\omega}_k)$.
3. If $\overline{y}_{m_k}(\overline{X}_{k-1};\underline{\omega}_k) \geq \overline{y}_{m_k}(\overline{X}_{k-1}+\varepsilon_k;\underline{\omega}_k)$ then set $Z_{n-1} = \overline{X}_{k-1}, Z_n = \overline{X}_{k-1}+\varepsilon_k$; otherwise set $Z_{n-1} = \overline{X}_{k-1}+\varepsilon_k, Z_n = \overline{X}_{k-1}$.
4. Set $t = \text{sign}(Z_n - Z_{n-1}), Z_{n-2} = Z_{n-1}$.
5. Find points that satisfy conditions C.1, C.3, and C.4: repeat Steps 5(a)–(d) until $Z_{n-2} < Z_{n-1} < Z_n$, $\overline{y}_{m_k}(Z_{n-2};\underline{\omega}_k) \geq \overline{y}_{m_k}(Z_{n-1};\underline{\omega}_k)$ and $\overline{y}_{m_k}(Z_n;\underline{\omega}_k) \geq \overline{y}_{m_k}(Z_{n-1};\underline{\omega}_k)$.
   (a) Set $Z_{n+1} = Z_n + tc_2|Z_n - Z_{n-1}|$.
   (b) Set $Z_{n-2} = Z_{n-1}, Z_{n-1} = Z_n, Z_n = Z_{n+1}$.
   (c) Simulate at Z_n to get $\overline{y}_{m_k}(Z_n;\underline{\omega}_k)$.
6. Initialize $X_a = Z_{n-2}, X_b = Z_{n-1}, X_c = Z_n$.
7. Find points X_a, X_b, X_c that satisfy conditions C.1, C.2, C.3, and C.4: repeat Steps 7(a)–(f) until $|X_c - X_a| \leq \varepsilon_k$.
   (a) Set $Z = (X_b + X_c)/2$.
   (b) Simulate at Z to obtain $\overline{y}_{m_k}(Z;\underline{\omega}_k)$.
   (c) If $\overline{y}_{m_k}(Z;\underline{\omega}_k) \leq \overline{y}_{m_k}(X_b;\underline{\omega}_k)$, then set $X_a = X_b, X_b = Z$. Otherwise set $X_c = Z$.
   (d) Set $Z = (X_a + X_b)/2$.
   (e) Simulate at Z to obtain $\overline{y}_{m_k}(Z;\underline{\omega}_k)$.
   (f) If $\overline{y}_{m_k}(Z;\underline{\omega}_k) \leq \overline{y}_{m_k}(X_b;\underline{\omega}_k)$, then set $X_c = X_b, X_b = Z$. Otherwise set $X_a = Z$.
8. Compute X_k as the minimizer of the quadratic passing through the three points $(X_a, \overline{y}_{m_k}(X_a;\underline{\omega}_k))$,

$(X_b, \overline{y}_{m_k}(X_b;\underline{\omega}_k))$, and $(X_c, \overline{y}_{m_k}(X_c;\underline{\omega}_k))$.
If $\overline{y}_{m_k}(X_a;\underline{\omega}_k) = \overline{y}_{m_k}(X_b;\underline{\omega}_k) = \overline{y}_{m_k}(X_c;\underline{\omega}_k)$, then set $X_k = X_a$.

9. Set $\overline{X}_k = \sum_{j=1}^{k} m_j X_j / \sum_{j=1}^{k} m_j$ and go to Step 1.

6 CONCLUDING REMARKS AND FUTURE RESEARCH

This paper addresses the question of choosing *parameters* in RA algorithms, where the term *parameters* is broadly used. Specifically, the paper identifies the rate at which the sequence of error tolerances $\{\varepsilon_k\}$ should be decreased in RA algorithms in order to achieve optimal convergence rate of the retrospective solutions $\{X_k\}$ to the true solution x^*. The results, in loose terms, suggest that it is best to choose the error tolerances so that they go to zero at the same rate at which solutions to the sample-path problems converge to the true solution. Since the latter rate is usually decided by the rate at which the sample sizes used to generate the sample-path problems go to ∞, the results point to the relative rates at which $\{\varepsilon_k\}$ and $\{m_k\}$ should be going to their respective limits to ensure optimal convergence.

The iteration-level termination criterion used in this paper is $\|X_k - X_k^*\| \leq \varepsilon_k$. So, the kth iteration is terminated when a solution X_k, within a distance ε_k from some minimizer of the objective function generated during the kth iteration, is identified. Although the results presented in this paper assume that the above termination criterion is in effect, parallel results can be derived in a similar fashion for most other termination criteria that one sees in the literature.

Several interesting directions can be pursued in future research. We list three that we consider important.

(i) In RA algorithms, frequently, it is impossible to verify if a retrospective solution X_k satisfies a specified termination criterion. In deterministic nonlinear programming, this problem is addressed heuristically or using statistical inference. The latter approach may be useful even in the RA context but an important question is the choice of the Type I error α_k if a hypothesis testing procedure is used. The value of α_k will decide the trade-off between the amount of computing to be done during the kth iteration and the required level of certainty on the satisfaction of the termination criterion.

(ii) The relation between $\{m_k\}$ and $\{\varepsilon_k\}$ is derived assuming i.i.d sampling in generating the sample-path function $\overline{y}_{m_k}(x;\underline{\omega}_k)$. It is likely that similar results hold even more generally, and characterizing these more general conditions is interesting.

(iii) The feasible set $\mathfrak{D}$ is assumed to be fixed and known in this paper. This is frequently not the case. See, for example, Atlason, Epelman, and Henderson (2002) and Nemirovski and Shapiro (2004). Investigating the choice of parameters in contexts where the feasible space itself needs to be estimated is an interesting problem.

REFERENCES

Andradóttir, S. 1990. Stochastic optimization with applications to discrete event systems. Ph. D. thesis, Department of Operations Research, Stanford University, Stanford, CA.

Andradóttir, S. 1991. A projected stochastic approximation algorithm. In *Proceedings of the 1991 Winter Simulation Conference*, ed. B. L. Nelson, D. W. Kelton, and G. M. Clark, 854–957.

Atlason, J., M. A. Epelman, and S. G. Henderson. 2002. Call center staffing with simulation and cutting plane methods. *Annals of Operations Research* 127:333–358.

Bazaara, M. S., H. Sherali, and C. M. Shetty. 2006. *Nonlinear programming: Theory and algorithms*. New York, NY.: John Wiley & Sons.

Bertsekas, D. 1999. *Nonlinear programming*. Nashua, NH.: Athena Scientific.

Blum, J. 1954. Multidimensional stochastic approximation. *Annals of Mathematical Statistics* 25:737–744.

Chen, H., and B. W. Schmeiser. 2001. Stochastic root finding via retrospective approximation. *IIE Transactions* 33:259–275.

de Mello, T. H. 2003. Variable-sample methods for stochastic optimization. *ACM Transactions on Modeling and Computer Simulation* 13:108–133.

Dupačová, J., and R. J. B. Wets. 1988. Asymptotic behavior of statistical estimators and of optimal solutions of stochastic optimization problems. *The Annals of Statistics* 16:1517–1549.

Fabian, V. 1968. On asymptotic normality in stochastic approximation. *Annals of Mathematical Statistics* 39:1327–1332.

Fu, M. C. 1994. Optimization via simulation: a review. *Annals of Operations Research* 53:199–247.

He, Y., M. C. Fu, and S. Marcus. 2003. Convergence of simultaneous perturbation stochastic approximation for nondifferentiable optimization. *IEEE Transactions on Automatic Control* 48:1459–1463.

Healy, K., and L. W. Schruben. 1991. Retrospective simulation response optimization. In *Proceedings of the 1991 Winter Simulation Conference*, ed. B. L. Nelson, D. W. Kelton, and G. M. Clark, 954–957: Institute of Electrical and Electronics Engineers: Piscataway, New Jersey.

Kesten, H. 1958. Accelerated stochastic approximation. *Annals of Mathematical Statistics* 21:41–59.

Kiefer, J., and J. Wolfowitz. 1952. Stochastic estimation of the maximum of a regression function. *Annals of Mathematical Statistics* 23:462–466.

Kushner, H., and D. Clark. 1978. *Stochastic approximation methods for constrained and unconstrained systems*. New York, NY.: Springer-Verlag.

Nemirovski, A., and A. Shapiro. 2004. Convex approximations of chance constrained programs. *Optimization Online*. <http://www.optimization-online.org/>.

Plambeck, E. L., B. R. Fu, S. M. Robinson, and R. Suri. 1996. Sample-path optimization of convex stochastic performance functions. *Mathematical Programming* 75:137–176.

Robbins, H., and S. Monro. 1951. A stochastic approximation method. *Annals of Mathematical Statistics* 22:400–407.

Rubinstein, R. Y., and A. Shapiro. 1993. *Discrete event systems: sensitivity analysis and stochastic optimization by the score function method*. New York, NY.: John Wiley & Sons.

Ruszczynski, and Shapiro. (Eds.) 2003. *Stochastic programming. Handbook in operations research and management science*. New York, NY.: Elsevier.

Safizadeh, M. H. 1990. Optimization in simulation: current issues and the future outlook. *Naval Research Logistics* 37:807–825.

Shapiro, A. 2000. Stochastic programming by monte carlo simulation methods. *Stochastic Programming E-Print Series*. <http://hera.rz.hu-berlin.de/speps/>.

Shapiro, A., and T. H. de Mello. 2000. On the rate of convergence of optimal solutions of monte carlo approximations of stochastic programs. *SIAM Journal on Optimization* 11 (1): 70–86.

Spall, J. C. 1998. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems* 34:817–823.

Spall, J. C. 2000. Adaptive stochastic approximation by the simultaneous perturbation method. *IEEE Transactions on Automatic Control* 45:1839–1853.

Venter, H. J. 1967. An extension of the robbins-monro procedure. *Annals of Mathematical Statistics* 38:181–190.

Wasan, M. T. 1969. *Stochastic approximation*. Cambridge, UK: Cambridge University Press.

AUTHOR BIOGRAPHY

RAGHU PASUPATHY is an assistant professor in the Industrial and Systems Engineering department at Virginia Tech. His current research interests include simulation-optimization, stochastic root finding and Monte Carlo methods. He is a member of INFORMS and IIE.