http://pubsonline.informs.org/journal/opre/

OPERATIONS RESEARCH
Articles in Advance, pp. 1–15
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Single Observation Adaptive Search for Continuous Simulation Optimization

Seksan Kiatsupaibul,[a] Robert L. Smith,[b] Zelda B. Zabinsky[c]

[a] Department of Statistics, Chulalongkorn University, Bangkok 10330, Thailand; [b] Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109; [c] Department of Industrial and Systems Engineering, University of Washington, Seattle, Washington 98195

Contact: seksan@cbs.chula.ac.th, https://orcid.org/0000-0002-2075-9183 (SK); rlsmith@umich.edu, https://orcid.org/0000-0002-4669-8948 (RLS); zelda@u.washington.edu, https://orcid.org/0000-0003-1838-4981 (ZBZ)

Received: April 18, 2016
Revised: June 19, 2017; February 2, 2018; February 26, 2018
Accepted: March 16, 2018
Published Online in Articles in Advance: December 12, 2018

Subject Classifications: nonlinear programming: algorithms; simulation: efficiency; statistics: sampling
Area of Review: Simulation

https://doi.org/10.1287/opre.2018.1759

Abstract. Optimizing the performance of complex systems modeled by stochastic computer simulations is a challenging task, partly because of the lack of structural properties (e.g., convexity). This challenge is magnified by the presence of random error whereby an adaptive algorithm searching for better designs can at times mistakenly accept an inferior design. In contrast to performing multiple simulations at a design point to estimate the performance of the design, we propose a framework for adaptive search algorithms that executes a *single* simulation for each design point encountered. Here the estimation errors are reduced by averaging the performances from previously evaluated designs drawn from a shrinking ball around the current design point. We show under mild regularity conditions for continuous design spaces that the accumulated errors, although dependent, form a martingale process, and hence, by the strong law of large numbers for martingales, the average errors converge to zero as the algorithm proceeds. This class of algorithms is shown to converge to a *global* optimum with probability one. By employing a shrinking ball approach with single observations, an adaptive search algorithm can simultaneously improve the estimates of performance while exploring new and potentially better design points. Numerical experiments offer empirical support for this paradigm of single observation simulation optimization.

Funding: Financial support was provided by the National Science Foundation [Grant CMMI-1632793].
Supplemental Material: The e-companion is available at https://doi.org/10.1287/opre.2018.1759.

Keywords: simulation optimization • adaptive search • martingale processes

1. Introduction

Stochastic optimization problems where an objective function is noisy and must be estimated are finding applications in diverse areas, spanning engineering, economics, computer science, business, and biological science. Simulation optimization algorithms that integrate search for the optimum with observations from a noisy objective function have been proposed in both continuous and discrete domains (Pasupathy and Ghosh 2013, Fu 2015).

Striking a balance between *exploration* of new points and *estimation* of potentially good points is critical for computationally efficient algorithms. We present a *class* of adaptive search algorithms that perform exactly *one* simulation per design point, which we call *single observation search algorithms* (SOSA). This class of SOSA algorithms combines exploration with estimation by estimating the expectation of the objective function at a point with an average of observed values from previously visited *nearby* points. The nearby points are within a shrinking ball around the current point. The challenge is to ensure that the errors associated with the estimates of the expectation of the noisy objective function do not bias the adaptive search algorithm and potentially lead to mistakenly accepting inferior solutions. We prove convergence to a *global* optimum for this class of SOSA algorithms under some mild regularity conditions. Then any adaptive search algorithm that fits into this class can utilize single observations within shrinking balls in contrast to multiple repetitions at a point to successfully search for a global optimum.

The idea of simulating a single observation per design point was first used by Robbins and Monro for estimating gradients in their classic stochastic approximation algorithm (see, for instance, Robbins and Monro 1951, Kushner and Yin 2003, and Chau and Fu 2015). This class of stochastic approximation algorithms have proven to be very successful at optimizing noisy functions on continuous domains (see Chau and Fu 2015). These classic stochastic approximation algorithms are based on steepest descent and are shown to converge to a *local* optimum. We consider the question of whether there exists single observation per design point algorithms that converge to a *global* optimum. The insight of Robbins and Monro in their classic stochastic

approximation algorithm with single observations, is "... if the step sizes in the parameter updates are allowed to go to zero in an appropriate way as (iterations go to infinity) then there is an *implicit averaging* that eliminates the effects of the noise in the long run." (Kushner and Yin 2003, p. viii). The random error associated with the steepest descent type method can be shown to be a martingale difference. This martingale property can be used to prove the convergence to a *local* optimum of the class of the steepest descent type methods. This paper shows how to extend implicit averaging in a way that provides convergence to a *global* optimum for our class of SOSA algorithms on nonconvex, multimodal, noisy optimization problems.

A necessary characteristic of our class of SOSA algorithms is that the underlying adaptive algorithm converges to a global optimal value when the objective function is *not* noisy. There are many such adaptive search algorithms for global optimization, for example, simulated annealing, evolutionary algorithms, model-based algorithms (Hu et al. 2007, 2014), nested partition method (Shi and Ólafsson 2000a), and interacting particle algorithms (Molvalioglu et al. 2009, 2010; Mete and Zabinsky 2014). (See Zabinsky 2011 for an overview.) We characterize algorithms in SOSA by their sampling distributions for generating sequential points. Because many adaptive search algorithms for global optimization maintain a sampling distribution that is bounded away from zero on the feasible region, we prove that this condition is sufficient to satisfy the assumptions of the convergence analysis and, consequently, that such an algorithm converges to a true global optimum using single observations with the shrinking ball approach.

The challenge in proving convergence to a global optimum with the additional demand of estimation is due to the complex *dependencies* among the observational errors introduced by an adaptive algorithm. An adaptive algorithm is influenced by past errors as it seeks new candidates and thus may be biased toward design points that were observed to be better than their true value. We show in this paper that underlying martingale properties exhibited by these adaptive algorithms can nonetheless be used to prove convergence to a global optimum.

Baumert and Smith (2002) introduced the idea of estimating the objective function at a specific design point x by including other points within a shrinking ball around x, thus never repeating a simulation at a single design point. Their algorithm was based on pure random search, which generates points independently and thus is not adaptive thereby avoiding dependencies among subsequent errors.

Andradóttir and Prudius (2010) investigated three methods, namely, Adaptive Search with Resampling (ASR), deterministic shrinking ball, and stochastic shrinking ball. They proved that the three algorithms are strongly convergent to a global optimum, but again, the deterministic and the stochastic shrinking ball approaches are based on pure random search with independent sampling.

Closely related works in the area of simulation optimization include: Stochastic Approximation (Spall 2003; Borkar 2008, 2013), Sample Average Approximation (SAA) (Kleywegt et al. 2002, Kim et al. 2015), Gradient-based Adaptive Stochastic Search for Simulation Optimization (Zhou et al. 2014), Nested Partitions Method (Shi and Ólafsson 2000b), and Low-Dispersion Point Sets (Yakowitz et al. 2000). These works either employ a gradient-based approach, exploit special structure of the feasible region, or use multiple observations of sampled design points. Stochastic approximation (Robbins and Monro 1951; Kushner and Yin 2003; Spall 2003; Borkar 2008, 2013; Chau and Fu 2015; Kim et al. 2015), starting with the seminal work of Robbins and Monro (Robbins and Monro 1951), is widely used for continuous problems with many applications. It is a first-order method with single or multiple observations per point that typically converges to a first-order stationary point. In contrast, SOSA does not require gradient information, allows a general continuous feasible region, and uses a single observation per sampled design point. Sample average approximation (Kleywegt et al. 2002, Homem-De-Mello 2003, Kim et al. 2015) has been used in both continuous and discrete stochastic optimization. SAA takes a collection of draws of the random vector U (e.g., $u_1, \ldots, u_N$) that may be independent identically distributed draws or possibly quasi-Monte Carlo samples or Latin hypercube designs and then solves an associated problem, $\min_{x \in S} \hat{f}(x)$, where $\hat{f}(x) = (1/N)\sum_{i=1}^{N} g(x, u_i)$. Under certain conditions, SAA converges to the global optimum as $N \to \infty$. Retrospective approximation (Pasupathy 2010) and variable sample size methods (Homem-De-Mello 2003) are efficient variations of SAA. SAA is also related to scenario-based stochastic programming (Shapiro et al. 2009), where the random draws are performed before the optimization over x is performed. Other simulation optimization methods (ours included) consider each x and then draw from U. Both methods have complex correlations and dependencies.

Meta-models (Barton and Meckesheimer 2006, Pedrielli and Ng 2015) are also commonly used in simulation optimization. SOSA, in principle, can be viewed as a meta-model and resembles Kriging (Pedrielli and Ng 2015) in that the objective function at nonsampled design points are estimated based on weighted observations of the function at other sampled design points. However, Kriging assumes conditions that are different from ours.

In this paper, we consider a broad class of adaptive random search algorithms in a continuous domain and prove that the accumulated error of the search process looking forward from the current candidate point follows a martingale process. Because the errors are zero in expectation for each point, the average error converges to zero from the strong law of large numbers for martingales (see "the impossibility of systems" in the work of Feller 1971). This allows us to prove convergence to a global optimum in probability for a broad class of adaptive random search algorithms with mild assumptions and using a single observation per design point.

We provide numerical results using two algorithms; both algorithms are run with single observations (SOSA), as well as with multiple replications implemented with ASR as by Andradóttir and Prudius (2010). The two algorithms are (1) a sampler based on a modified version of Improving Hit-and-Run (IHR) (see Zabinsky et al. 1993, Zabinsky and Smith 2013) and (2) a uniform local/global sampler originally used by Andradóttir and Prudius (2010). Andradóttir and Prudius (2010) demonstrated that ASR performed better computationally than nonadaptive shrinking ball approaches in high dimensions. In this paper, we demonstrate on two test problems in 10 dimensions that SOSA performs better computationally than ASR for both sampling methods tested. This is an encouraging result, that implicit averaging over shrinking balls can be very effective and can improve computational performance of stochastic searches in general. The averaging of nearby points seems to have the effect of discovering local and global trends in the objective function, thus enabling an adaptive algorithm to sample effectively.

2. Single Observation Simulation Optimization

The stochastic optimization problem we consider is

$$\min_{x\in S} f(x), \tag{1}$$

where $x \in S \subset \Re^d$ and

$$f(x) = \mathbb{E}[g(x, U)]. \tag{2}$$

However, the objective function $f(x)$ cannot be evaluated exactly. Instead, we have a noisy evaluation available; that is, the performance at a design point $x \in S \subset \Re^d$ is given by $g : S \times \Omega \to \Re$, where U is a random element over a probability space denoted $(\Omega, \mathcal{A}, \mathbb{P})$. In discrete-event simulation, randomness is typically generated from a sequence of pseudo-random numbers, therefore, we let U represent an independent and identically distributed (i.i.d.) sequence of uniform random variables. We assume that f is continuous and S is compact so that a minimum exists. Let $\mathcal{X}^* = \arg\min_{x\in S} f(x)$ denote the set of optimal solutions, and let f^* be the optimal value.

A common approach in simulation optimization is to estimate $f(x)$ by observing the output of a simulation run, $g(x, u)$, where u is a realization of the random variable U. The difference between the observed performance and mean performance, denoted

$$Z(x) = g(x, U) - f(x), \tag{3}$$

represents the random observational error.

When the random observational errors are independent across all iterations of the algorithm, and identically distributed, then the strong law of large numbers can be invoked to prove the error goes to zero as iterations increase to infinity. The challenge with establishing convergence to optimality for an *adaptive* algorithm for global optimization is that the random errors are, in general, neither identically distributed nor independent. An *adaptive* algorithm that favors "better" design points introduces complex dependencies among the errors. Also because points that appear "better" influence the adaptive algorithm, the optimal value estimates tend to be negatively biased.

Example: Illustration of Dependent Estimation Errors. Suppose S is the union of two nonoverlapping balls we will call ball L and ball R. Moreover, suppose that the objective function values $f(x)$, for $x \in L$, are better (less) than those in R. Suppose in the initial step of an adaptive algorithm we begin by sampling a point from ball L and we observe its objective function value. Next we sample a point from the other ball R and compare its value with that of our point in L. The third point will be sampled from the ball with the smaller observed value.

Suppose that the error associated with the first noisy observation in L is negative, i.e., the observed value is smaller than the true $f(x)$. Now suppose the third point is in R. In this case, a negative error at the first point sampled means the error at the second point must also be negative because its expected value is inferior to the first point. This example illustrates that there is a dependency between the errors from the first and the second observations. In general, there are subtle dependencies that can be induced by adaptive search algorithms.

In Section 3, we show that, whereas errors looking backward from the current iteration point are dependent (e.g., looking at the first and second points, having sampled the third), errors looking forward when conditioning on the identity of the current iteration point (e.g., looking at the fourth point, having sampled the third) are independent of past errors.

In this paper, we rigorously analyze the accumulated error associated with a class of adaptive random search

algorithms. As suggested in the example above, while the accumulated error of the *entire* process does **not** form a martingale, the accumulated error of the process *after* a point has been evaluated **does** form a martingale. This insight allows us to prove convergence to a global optimum with probability one for a broad class of adaptive random search algorithms with mild assumptions and using a single observation per design point. The key is to "slow down" the search so that it does not converge prematurely to a wrong solution.

We make the following three assumptions regarding the problem. These assumptions are relatively mild and are typically satisfied in most continuous simulation optimization problems.

Assumption 1. *The feasible set $S \subset \Re^d$ is a closed and bounded convex set with nonempty interior.*

Assumption 2. *The objective function $f(x)$ is continuous on S.*

We consider two versions of the next assumption, namely, Assumption 3 and Assumption 3′. Assumption 3 requires that the random error be bounded and is satisfied by many bounded distributions. However, it does not include distributions having infinite support, such as normal or gamma distributions. Assumption 3′ requires that the random error has bounded variance, and the normal and gamma distributions with finite variance satisfy Assumption 3′. In the next section, we see that Assumption 3 leads to a stronger convergence result (convergence with probability one) than Assumption 3′ (convergence in probability).

Assumption 3. *The random error $(g(x, U) - f(x))$ is uniformly bounded over $x \in S$; that is, there exists $0 < \alpha < \infty$ such that, for all $x \in S$, with probability one,*

$$|g(x, U) - f(x)| < \alpha.$$

Assumption 3′. *The random error $(g(x, U) - f(x))$ has bounded variance over $x \in S$.*

3. A Class of Adaptive Random Search Algorithms and Convergence Analysis

We propose a class of adaptive stochastic search algorithms with single observations per design for the continuous simulation optimization problem in Equation (1). We show that, under some regularity conditions, an algorithm in this class will generate a sequence of objective function estimates that converge to the true global optimum with probability one.

In the course of the algorithm, design points are sequentially sampled from the design space S, according to an adaptive sampling distribution, denoted by q_n at iteration n. The objective function at a point is estimated using the observed function values of nearby sample points within a certain radius. The use of nearby points allows for a strong law of large numbers type of convergence by asymptotically eliminating the random error associated with the sequence of observations. The radius shrinks as the algorithm progresses, hence the image of shrinking balls. This shrinking process asymptotically eliminates the systematic bias of using estimates of neighboring points because the objective function $f(x)$ is continuous in x.

For each $x \in S$, let $B(x, r)$ be the ball centered at x with radius r. Let $\mathcal{X}_n$ and $\mathcal{Y}_n$ be, respectively, the set of sample points obtained in the course of the Single Observation Search Algorithms and their corresponding function evaluations up to iteration n. For $x_i \in \mathcal{X}_n$, the objective function estimate $\hat{f}_n(x_i)$ of x_i comes from the average of the function evaluations of the sample points that fall into the balls centered at x_i. Let $l_n(x_i)$ denote the number of sample points that fall into the balls centered at x_i, called contributions to the estimate of x_i.

Consider the following class of algorithms using the shrinking ball concept. Note that, to guarantee convergence to a true global optimum, regularity conditions for the sampling density q_n and the parameter sequences r_n and i_n are required. In what follows, let $|A|$ for a set A denote the number of elements in A.

3.1. Single Observation Search Algorithms (SOSA)

We are given

- A continuous initial sampling density for search on S, $q_1(x)$, and a family of continuous adaptive search sampling distributions on S with density

$$q_n(x \mid x_1, y_1, \ldots, x_{n-1}, y_{n-1}), \quad n = 2, 3, \ldots,$$

where x_n is the sample point at iteration n and y_n is its observed function value.

- A sequence of radii $r_n > 0$.
- A sequence $i_n < n$.

Step 0: Sample x_1 from q_1, observe $y_1 = g(x_1, u_1)$ from the simulation, where u_1 is a sample value having the same distribution as U and independent of x_1. Set $\mathcal{X}_1 = \{x_1\}$ and $\mathcal{Y}_1 = \{y_1\}$. Also, set $\hat{f}_1(x_1) = \hat{f}_1^*(x_1) = y_1$, $l_1(x_1) = 1$, and $x_1^* = x_1$. Set $n = 2$.

Step 1: Given $x_1, y_1, \ldots, x_{n-1}, y_{n-1}$, sample the next point, x_n, from q_n. Independent of $x_1, y_1, \ldots, x_{n-1}, y_{n-1}, x_n$, obtain a sample value u_n having the same distribution as U and evaluate the objective function value $y_n = g(x_n, u_n)$.

Step 2: Update $\mathcal{X}_n = \mathcal{X}_{n-1} \cup \{x_n\}$ and $\mathcal{Y}_n = \mathcal{Y}_{n-1} \cup \{y_n\}$. For each $x \in \mathcal{X}_n$, update the contribution and the estimate of the objective function value as

$$l_n(x) = |\{k \le n : x_k \in B(x, r_k)\}| = \begin{cases} l_{n-1}(x) & \text{if } x_n \notin B(x, r_n) \\ l_{n-1}(x) + 1 & \text{if } x_n \in B(x, r_n), \end{cases} \tag{4}$$

and

$$\hat{f}_n(x) = \frac{\sum_{\{k\le n : x_k \in B(x,r_k)\}} y_k}{|\{k \le n : x_k \in B(x, r_k)\}|} = \begin{cases} \hat{f}_{n-1}(x), & \text{if } x_n \notin B(x, r_n), \\ ((l_n(x)-1)\hat{f}_{n-1}(x) + y_n)/l_n(x), & \text{if } x_n \in B(x, r_n), \end{cases} \tag{5}$$

where $B(x, r_k)$ is a ball of radius r_k centered at x. Estimate the optimal value as

$$\hat{f}_n^* = \min_{x\in\mathcal{X}_{i_n}} \hat{f}_n(x), \tag{6}$$

and estimate the optimal solution as

$$x_n^* \in \{x \in \mathcal{X}_{i_n} : \hat{f}_n(x) = \hat{f}_n^*\}, \tag{7}$$

where $\mathcal{X}_{i_n}$ is the subset of $\mathcal{X}_n$ that only includes points 1 through i_n.

Step 3: If a stopping criterion is met, stop. Otherwise, update $n \leftarrow n+1$ and go to Step 1.

Observe that the objective function estimate $\hat{f}_n(x)$ is defined for all $x \in S$. However, in the course of the algorithm, we only compute the estimate for $x \in \mathcal{X}_n$. To do this, the simplest way is to go through $\mathcal{X}_n$ once in each iteration and update the objective function estimates through the recursive formula in Equation (5). By doing so, up to iteration n, the estimates at sample points $x_1, x_2, \ldots, x_n$ will be updated $n, n-1, \ldots, 1$ times, respectively, which will result in at most $n(n+1)/2$ updates. At each iteration, the function estimate of a sample point will get updated only when the new sample point falls relatively close to a previously sampled point (within its ball of radius r_k). Therefore, up to iteration n, the function estimates will be calculated less than $n(n+1)/2$ times.

Notice that the algorithm takes the estimate of the optimal value on the nth iteration, $\hat{f}_n^*(x)$, not from all n function estimates, but from a subset of the function estimates up to i_n. By slowing the sequence of estimated values using i_n, we are able to ensure convergence of the optimal value estimate $\hat{f}_n^*(x)$ to the true optimal value f^*. The idea is that the shrinking balls around points used to estimate a global optimum shrink slowly enough to allow for the number of points in those balls to grow to infinity.

The main result of the paper is stated in Theorem 3, where we prove that an adaptive random search algorithm with single observations per design, using a shrinking ball approach, converges to a global optimum with probability one. The following corollary with the relaxed Assumption 3′ proves convergence to a global optimal value in probability. The convergence analyses rest on the martingale property of the random error, which we establish in Theorem 1.

To develop this martingale property, we investigate the random error at a design vector, as in Equation (3), $Z(x) = g(x, U) - f(x)$, and because

$$\mathbb{E}[Z(x)] = \mathbb{E}[g(x, U)] - f(x) = f(x) - f(x) = 0, \quad \text{for all } x \in S, \tag{8}$$

$Z(x)$ is a random error with zero expectation. It is worth noting that, rewriting Equation (3) as

$$g(x, U) = f(x) + Z(x) \tag{9}$$

now states that, given a design point x, a random performance can always be decomposed into a sum of its expected performance and a random error with zero expectation.

To establish the convergence result for SOSA, it is necessary that the sequence of random numbers used as input to the simulation are independent of the past information. To state this more formally, first, let X_n and Y_n denote the sample point and its corresponding objective function evaluation at iteration n, for $n = 1, 2, \ldots$. Then

$$Y_n = g(X_n, U_n), \tag{10}$$

where $\{U_n, n = 1, 2, \ldots\}$ are random elements, i.i.d. and have the same distribution as U. Because X_n, U_n, and Y_n are generated sequentially, we can construct a filtration, starting with $\mathcal{F}_0 = \sigma(X_1)$, the σ-field generated by X_1, and, for $n = 1, 2, \ldots$, $\mathcal{F}_n = \sigma(X_1, U_1, \ldots, X_n, U_n, X_{n+1})$, the σ-field generated by $X_1, U_1, \ldots, X_n, U_n, X_{n+1}$. Observe that X_n is $\mathcal{F}_{n-1}$ measurable. Because Y_n is a function of X_n and U_n, Y_n is $\mathcal{F}_n$ measurable. The process of (X_n, Y_n) is then adapted to the filtration $\{\mathcal{F}_n\}_{n=0}^{\infty}$. It is crucial to the convergence results that U_n is generated so that it is independent of $\mathcal{F}_{n-1}$.

We next establish that the expected random error conditioned on the filtration is zero. This property will induce a martingale process of accumulated errors and, finally, enable the optimal value estimates generated by the algorithm, $\{\hat{f}_n^*\}$, to converge to the true optimal value f^*.

Define the random error at iteration n by Z_n, where

$$Z_n = Y_n - f(X_n). \tag{11}$$

Note that Z_n, as well as Y_n, depend on X_n and U_n, but we suppress the arguments in the notation to simplify the presentation.

We now establish a crucial martingale property, that $\mathbb{E}[Z_n \mid \mathcal{F}_{n-1}] = 0$. Because X_n is $\mathcal{F}_{n-1}$ measurable and U_n is independent of $\mathcal{F}_{n-1}$,

$$\mathbb{E}[Y_n \mid \mathcal{F}_{n-1}] = \mathbb{E}[g(X_n, U_n) \mid \mathcal{F}_{n-1}] = \mathbb{E}[g(X_n, U_n) \mid X_n],$$

and because U_n is identically distributed as U,

$$= \mathbb{E}[g(X_n, U) \mid X_n] = f(X_n). \tag{12}$$

Again, because X_n is $\mathcal{F}_{n-1}$ measurable,

$$\begin{aligned}\mathbb{E}[Z_n \mid \mathcal{F}_{n-1}] &= \mathbb{E}\big[Y_n - f(X_n) \mid \mathcal{F}_{n-1}\big] \\ &= \mathbb{E}[Y_n \mid \mathcal{F}_{n-1}] - f(X_n) = 0. \end{aligned} \tag{13}$$

The result in Equation (13) provides the basis of the martingale property because the error on iteration n, conditioned on the past information in $\mathcal{F}_{n-1}$, is zero.

Furthermore,

$$\mathbb{E}[Z_n] = \mathbb{E}[\mathbb{E}[Z_n \mid \mathcal{F}_{n-1}]] = \mathbb{E}[0] = 0, \tag{14}$$

which establishes that, according to Equations (12) and (14), we can decompose Y_n into its conditional expectation term and its error term,

$$Y_n = f(X_n) + Z_n, \tag{15}$$

where the random error Z_n has zero expectation.

Now we express the accumulated error in estimating $f(X_i)$ associated with the sample point X_i in terms of the error coming from the points in the balls around X_i.

At iteration n, and for a fixed sample point X_i, for $i \le n$, let $M_n(X_i)$ be the accumulated error in estimating $f(X_i)$ using the function evaluations from the points X_k, $k = 1, \ldots, n$ that fall into balls around X_i. We define an indicator function to identify the sample points in balls around X_i,

$$I_k(X_i) = \begin{cases} 1 & \text{if } X_k \in B(X_i, r_k) \\ 0 & \text{if } X_k \notin B(X_i, r_k), \end{cases}$$

for $k = 1, \ldots, n$. Using the indicator function, we have

$$M_n(X_i) = \sum_{k=1}^{n} I_k(X_i) Z_k. \tag{16}$$

It is important to note that $\{M_n(X_i), n = 1, 2, \ldots\}$ for $i > 1$ is not a martingale, owing to the dependencies on early sample points in the sequence. In fact, when $n < i$, $M_n(X_i)$ depends on unrealized information of X_i. To be exact, $M_n(X_i)$, where $n < i$, is not $\mathcal{F}_n$ measurable.

At each iteration, the best candidate X_n^* is chosen as the point whose function estimate is the smallest so far. Because the function estimate is formed by the function observations of previous points in the ball around X_n^*, these objective function observations tend to be negatively biased. The errors from these points tend to be concurrently negative and, hence, are correlated.

We have discovered that we can decompose the accumulated error into two parts (the error from the sample points that preceded X_i and the error from the sample points that were sampled after X_i) which allows us to establish the martingale property for the second part of the error. We let

$$M_n^i(X_i) = \sum_{k=i}^{n} I_k(X_i) Z_k, \quad n = i, i+1, \ldots \tag{17}$$

be the accumulated error from function evaluations taken from iteration i onward to iteration n. Note that $M_n^i(X_i)$ is the sum of $(n - i + 1)$ error terms. Define $M_{i-1}^i(X_i) = 0$. Now, using Equations (16) and (17), we decompose $M_n(X_i)$ into two parts

$$M_n(X_i) = \sum_{k=0}^{i-1} I_k(X_i) Z_k + M_n^i(X_i), \tag{18}$$

where $I_0(X_i) Z_0 = 0$ as a convention.

In Theorem 1, we show that the accumulated error about a sample point X_i from function evaluations taken from iteration i onward to iteration n is a martingale.

Theorem 1. *For any i, $i = 1, 2, \ldots$, $\{M_n^i(X_i), n = i, i+1, \ldots\}$ is a martingale with respect to the filtration $\{\mathcal{F}_n, n = i, i+1, \ldots\}$.*

Proof. Fix i. Define

$$\tilde{M}_n^i = \sum_{k=i}^{n} Z_k$$

as the accumulated error from *all* points sampled on the iterations from iteration i through iteration n. We first show that $\{\tilde{M}_n^i, n = i, i+1, \ldots\}$ is a martingale with respect to the filtration $\{\mathcal{F}_n, n = i, i+1, \ldots,\}$. This is equivalent to showing that $\mathbb{E}[|\tilde{M}_n^i|] < \infty$ and $\mathbb{E}[\tilde{M}_n^i \mid \mathcal{F}_{n-1}] = \tilde{M}_{n-1}^i$ for all $n \ge i$. By Assumption 3, $\mathbb{E}[|Z_n|] < \alpha < \infty$. By the triangular inequality of the absolute value function, $\mathbb{E}[|\tilde{M}_n^i|] \le (n - i + 1)\alpha < \infty$. In addition,

$$\begin{aligned}\mathbb{E}[\tilde{M}_n^i \mid \mathcal{F}_{n-1}] &= \mathbb{E}[Z_n + \tilde{M}_{n-1}^i \mid \mathcal{F}_{n-1}] \\ &= \mathbb{E}[Z_n \mid \mathcal{F}_{n-1}] + \mathbb{E}[\tilde{M}_{n-1}^i \mid \mathcal{F}_{n-1}] \\ &= \tilde{M}_{n-1}^i. \end{aligned}$$

The last equation follows from Equation (13) and that $\tilde{M}_{n-1}^i$ is $\mathcal{F}_{n-1}$ measurable for all $n \ge i$. Therefore, $\{\tilde{M}_n^i, n = i,\ i+1, \ldots\}$ is a martingale.

Observe that $I_k(X_i)$ is $\mathcal{F}_{k-1}$ measurable for $k = i$, $i+1, \ldots$. Therefore, $I_k(X_i)$ is a decision function with respect to the filtration $\{\mathcal{F}_n, n = i, i+1, \ldots,\}$. Now, for $n = i, i+1, \ldots,$

$$M_n^i(X_i) = \sum_{k=i}^{n} I_k(X_i) Z_k = M_{n-1}^i(X_i) + I_n(X_i)(\tilde{M}_n^i - \tilde{M}_{n-1}^i).$$

Therefore, by *the impossibility of systems* (Feller 1971), $\{M_n^i(X_i), n = i, i+1, \ldots\}$ is a martingale. □

We now express the estimate of the function value at a sample point and the estimate of the optimal

value in terms of X_i, $f(X_i)$, and $M_n(X_i)$. For a fixed i, let $L_n(X_i)$ be the number of sample points that fall into the balls $B(X_i, r_k)$ around X_i where $k = 1, \ldots, n$ and $n \geq i$; that is,

$$L_n(X_i) = \sum_{k=1}^{n} I_k(X_i).$$

The estimate of the function value at a sample point X_i can be expressed as

$$\hat{f}_n(X_i) = \frac{\sum_{k=1}^{n} I_k(X_i) Y_k}{L_n(X_i)} = \frac{\sum_{k=1}^{n} I_k(X_i) f(X_k)}{L_n(X_i)} + \frac{M_n(X_i)}{L_n(X_i)}, \quad (19)$$

where the first term includes the systematic bias and the second term is the accumulated error. The systematic bias is created by the fact that an estimate of the function value at a point includes function value estimates of points around it with different expectations. The accumulated error $M_n(X_i)$ can be decomposed (see Equation (18)) into a nonmartingale accumulated error term, $\sum_{k=0}^{i-1} I_k(X_i) Z_k$, and a martingale process of the accumulated errors, $M_n^i(X_i)$, as a result of Theorem 1.

The estimate of the optimal value f^* is

$$\begin{aligned} \hat{f}_n^* &= \min_{i=1,\ldots,i_n} \{\hat{f}_n(X_i)\} \\ &= \min_{i=1,\ldots,i_n} \left\{ \frac{\sum_{k=1}^{n} I_k(X_i) f(X_k)}{L_n(X_i)} + \frac{M_n(X_i)}{L_n(X_i)} \right\}. \end{aligned} \quad (20)$$

Note that $\hat{f}_n^*$ is the minimum taken not from all n function estimates but from a subset of the function estimates up to i_n, where $i_n \leq n$. The size of the subset i_n is a control parameter required to ensure the convergence of the optimal value estimate $\hat{f}^*$ to the true optimal value f^* by slowing the search for an optimum.

Because $\hat{f}_n^*$, the estimate of the optimal value generated by SOSA, is taken from the minimum of a growing number of estimated function values to prove that $\hat{f}^*$, in fact, converges to f^*, the true optimal value, we require some form of uniformity in the convergence of these estimates. To establish the requirements, we add one more assumption.

Recall that $L_n(x)$ is the number of sample points that fall in the balls around x. Given a function of natural numbers $\tilde{L}(n)$, we define $D(n)$ to be the event that each design vector x has at least $\tilde{L}(n)$ sample points in the balls around x; that is,

$$D(n) = \bigcap_{x \in S} \{L_n(x) \geq \tilde{L}(n)\}.$$

The objective function evaluations of these sample points (one function evaluation per sample point) around a design point will form the estimate of the objective function of that design point.

The key idea is that the number of sample points in the balls around x grows at least as fast as $\tilde{L}(n)$ even though the radii of the balls are shrinking. In other words, the balls cannot shrink too quickly; the shrinking balls must maintain a threshold of sample points in them.

Definition 1. A function $h(n)$ is called $O(n^p)$, where $p \in \Re$ if there is a $0 < \kappa_r < \infty$ such that for all $n \in \mathbb{N}$, $0 \leq h(n) \leq \kappa_r n^p$. A function $h(n)$ is called $\Omega(n^p)$, where $p \in \mathbb{R}$ if there is a $0 < \kappa_L < \infty$ such that for all $n \in \mathbb{N}, h(n) \geq \kappa_L n^p$. A function $h(n)$ is called $\Theta(n^p)$ if it is both $O(n^p)$ and $\Omega(n^p)$.

Assumption 4. *Assume there exists* $1/2 < \gamma < 1$ *and a function* $\tilde{L}(n)$ *that is* $\Omega(n^\gamma)$ *such that*

$$\sum_{n=1}^{\infty} \mathbb{P}(D(n)^c) < \infty,$$

where $D(n)^c$ *is the complement of event* $D(n)$ *and* γ *is called an order of local sample density.*

Assumption 4 ensures that there are on the order of n^γ function evaluations used in the estimate of every point in the design space.

The single observation search algorithms can satisfy Assumption 4 if the sampling densities satisfy some regularity conditions, and if the radii of the balls do not shrink too quickly. In particular, if the search sampling density q_n, $n = 1, 2, \ldots$ is uniformly bounded away from zero on S and r_n is of $\Omega(n^{-(1-\gamma)/d})$, then Assumption 4 is satisfied. Many adaptive search algorithms for global optimization do have their search sampling density bounded away from zero, and in the next section we consider two such algorithms and prove that they satisfy Assumption 4 and hence converge to a global optimum.

In leading up to Theorem 3, we expand the estimate of the function value in Equation (19) as

$$\begin{aligned} \hat{f}_n(X_i) &= \frac{\sum_{k=1}^{n} I_k(X_i) f(X_i)}{L_n(X_i)} + \frac{\sum_{k=1}^{n} I_k(X_i)(f(X_k) - f(X_i))}{L_n(X_i)} \\ &\quad + \frac{\sum_{k=1}^{i-1} I_k(X_i) Z_k}{L_n(X_i)} + \frac{\sum_{k=i}^{n} I_k(X_i) Z_k}{L_n(X_i)} \\ &= f(X_i) + \left(\frac{\sum_{k=1}^{n} I_k(X_i) f(X_k)}{L_n(X_i)} - f(X_i) \right) \\ &\quad + \frac{\sum_{k=1}^{i-1} I_k(X_i) Z_k}{L_n(X_i)} + \frac{\sum_{k=i}^{n} I_k(X_i) Z_k}{L_n(X_i)}, \end{aligned} \quad (21)$$

to clearly identify the correct value in the first term, the bias due to nearby points in the second term, the nonmartingale accumulated error in the third term, and the martingale accumulated error in the fourth term. The second term, the bias, is created by the fact that an estimate of the function value at a point is formed by the function value of that point and other points around it but within the balls. On the basis of the continuity of

the objective function in Assumption 2, we employ Cesàro's Lemma (see Lemma EC.1 in the e-companion) with the shrinking ball mechanism to show that the bias term is washed away by averaging. The third term, the nonmartingale random error of the estimated objective function value of a specific sample point, is formed by the errors corresponding to other points that are sampled prior to that sample point. These nonmartingale random errors can be highly correlated and may not cancel each other out by averaging alone. However, it is considered a fixed term once a specific point has been sampled. Therefore, the slowing sequence, i_n, is employed to slow the growth of this term, causing this nonmartingale random error to diminish to zero when divided by the number of points in the associated balls. The fourth term, the martingale random error of the estimated objective function value of a specific sample point, is formed by the errors corresponding to points that are sampled from that specific sample point onward. The slowing sequence together with the martingale property through the Azuma–Hoeffding inequality (see Lemma EC.2 in the e-companion) causes the martingale random error to disappear. Thus, in the limit, we are left with the correct value for the estimate of the function value.

We next prove Theorem 2 showing that the probability that the estimate is incorrect by ε amount for the early portion of the estimates goes to zero as n goes to infinity, i.e.,

$$\lim_{n\to\infty} \mathbb{P}\left(\bigcup_{i=1}^{i_n} \{|\hat{f}_n(X_i) - f(X_i)| \geq \varepsilon\}\right) = 0.$$

For notational convenience, define $A(n,\varepsilon)$ as the event that, when we consider only the early portion of the sequence up to i_n, at least one objective function estimate is incorrect by more than the target error ε allowed for $\varepsilon > 0$; that is,

$$A(n,\varepsilon) = \bigcup_{i=1}^{i_n} \{|\hat{f}_n(X_i) - f(X_i)| \geq \varepsilon\}.$$

Theorem 2 gives conditions under which, the probability of missing this error target for the early portion of the estimates goes to zero as n goes to infinity. Theorem 2 makes use of the martingale property established in Theorem 1 and the slowing sequence i_n in SOSA.

Theorem 2. *If Assumptions 1, 2, 3, and 4 are satisfied, and if $i_n \uparrow \infty$ such that $i_n \leq n^s$, where $0 < s < \gamma$, then, for all $\varepsilon > 0$,*

$$\sum_{n=1}^{\infty} \mathbb{P}(A(n,\varepsilon)) < \infty.$$

Proof. See the e-companion. □

By Assumption 4, the algorithm generates sample points that fill the feasible region. Once the estimated objective function errors of all sample points are controlled as described in Theorem 2 and the objective function is continuous according to Assumption 2, the optimal value estimates converge to the true optimal value. This convergence property is formalized in Theorem 3.

Theorem 3. *If Assumptions 1, 2, 3, and 4 are satisfied, and if $i_n \uparrow \infty$ such that $i_n \leq n^s$, where $0 < s < \gamma$, then $\hat{f}_n^* \to f^*$ with probability one.*

Proof. See the e-companion. □

If Assumption 3 is relaxed to Assumption 3′, we have a weaker convergence in probability result.

Corollary 1. *If Assumptions 1, 2, 3′, and 4 are satisfied, and if $i_n \uparrow \infty$ such that $i_n \leq n^s$, where $0 < s < \gamma$, then, for all $\varepsilon > 0$,*

$$\lim_{n\to\infty} \mathbb{P}(|\hat{f}_n^* - f^*| \geq \varepsilon) = 0,$$

i.e., $\hat{f}_n^* \to f^*$ in probability.

Proof. See the e-companion. □

Note that Assumption 4 can also be relaxed but with a weaker convergence result as in Corollary 1. A weaker Assumption 4, requiring only that $\lim_{n\to\infty} \mathbb{P}(D(n)^c) = 0$, produces the same effect through the last term of Equation (EC.4), and thus, as in Corollary 2, $\hat{f}_n^*$ converges to f^* only in probability.

Now, in Corollary 2, we show that not only does SOSA converge to the optimal value, but, under appropriate conditions, it converges to an optimal solution. Let X_n^* represent the optimal solution estimate at iteration n. In the case of multiple optima, it is possible that the estimates jump within the set of optima, depending on the underlying sampling distributions q_n; however, the distance between the optimal solution estimate X_n^* and the set of global optima $\mathcal{X}^*$ converges to zero with probability one. For a solution $x \in \Re^d$ and a subset $E \subset \Re^d$, we define the distance from x to E as $\rho(x,E) = \inf_{y\in E} \|x - y\|$, where $\|\cdot\|$ denotes the Euclidean norm on $\Re^d$. When there is a unique optimum, the optimal solution estimate converges to the unique global optimum.

Corollary 2. *Suppose all the conditions in Theorem 3 are satisfied. Then, with probability one,*

$$\rho(X_n^*, \mathcal{X}^*) \to 0.$$

In addition, if there is a unique optimum, $\mathcal{X}^* = \{x^*\}$, then $X_n^* \to x^*$ with probability one.

Proof. See the e-companion. □

4. An Application: Single Observation Algorithms Based on Hit-and-Run

We compare four algorithms on two global optimization test problems with noisy objective functions.

The four algorithms comprise two from the SOSA framework and two from the ASR framework. Introduced by Andradóttir and Prudius (2010), ASR is an adaptive search framework that advocates performing repeated observations of objective function at sampled design points, which is in contrast to SOSA. Within each framework, we try two different samplers, one is the Improving Hit-and-Run (IHR) sampler and the other one is the sampler originally used with ASR in the work of Andradóttir and Prudius (2010), which we call the AP sampler. The four algorithms to be tested in this study are

1. SOSA with IHR sampler (IHR-SO);
2. SOSA with AP sampler (AP-SO);
3. ASR with IHR sampler (IHR-ASR);
4. ASR with AP sampler (AP-ASR).

The IHR-SO algorithm modifies the improving hit-and-run algorithm (IHR) (see Zabinsky et al. 1993, Zabinsky 2003, Ghate and Smith 2008, Zabinsky and Smith 2013) by incorporating the shrinking ball approach. The class of hit-and-run algorithms has a desirable property of efficiently converging to a target distribution with very mild assumptions (Smith 1984, Zabinsky and Smith 2013). IHR is a simplified version of Annealing Adaptive Search (AAS) algorithm (Romeijn and Smith 1994, Zabinsky 2003) with zero temperature. AAS is later shown to be approximated by Model-based Annealing Random Search (MARS) (Hu et al. 2007, 2014). Therefore, IHR can be considered a simple representative of a global optimization search engine in this class.

An early version of improving hit-and-run with single observation, but without shrinking balls, has been applied to a simulation optimization problem with encouraging computational results in Kiatsupaibul et al. (2015). A more extensive numerical study (Linz et al. 2017) showed a benefit of SOSA compared with multiple replications.

The AP sampler has been employed as the sampling strategy for ASR in the study of Andradóttir and Prudius (2010) with promising computational results. AP samples locally from the neighborhood of the current optimal solution estimate and also globally from the whole feasible region. In a sense, it is a simple version of the nested partitions methods (Shi and Ólafsson 2000a) that sample from a partition local to the current optimal solution estimate and its complement. Therefore, AP can be considered a simple representative of another class of global optimization search engine. The original combination of AP and ASR is called AP-ASR in this study. Here we also combine AP with SOSA, called AP-SO, and compare it against the original AP-ASR.

The IHR-SO and AP-SO algorithms share the property of having a positive probability of sampling anywhere in the space. Thus, as shown in Theorem 4 and Corollary 3, they satisfy the four assumptions for continuous optimization problems and converge to a global optimal value.

4.1. SOSA with IHR Sampler (IHR-SO)

Parameters: κ_r, starting ball radius; γ, order of local sample density; β, order of radius shrinkage; and s, order of slowing sequence. The parameters specify the following radii and slowing sequences:

$$\{r_n = \kappa_r n^{-\beta}, n = 1, 2, \ldots\} \quad \text{and} \quad \{i_n = \lfloor n^s \rfloor, n = 1, 2, \ldots\}.$$

Follow the steps of SOSA but replace Step 1 by the following IHR sampler.

Step 1: Given $x_1, y_1, \ldots, x_{n-1}, y_{n-1}$, let $\tilde{x}_{n-1} \in \arg\min_{x \in \mathscr{X}_{n-1}} \hat{f}_{n-1}(x)$. The IHR sampler obtains a new design point x_n by first sampling a direction v from the uniform distribution on the surface of a d-dimensional hypersphere, and second, sampling x_n uniformly distributed on the line segment Λ where $\Lambda = \{\tilde{x}_{n-1} + \lambda v : \lambda \in \Re\} \cap S$. Given x_n, sample u_n having the same distribution as U and evaluate $y_n = g(x_n, u_n)$.

4.2. SOSA with AP Sampler (AP-SO)

The same as IHR-SO but replace Step 1 by the following AP sampler.

Step 1: Given $x_1, y_1, \ldots, x_{n-1}, y_{n-1}$, let $\tilde{x}_{n-1} \in \arg\min_{x \in \mathscr{X}_{n-1}} \hat{f}_{n-1}(x)$. Sample from the AP sampler by sampling x_n uniformly on S with probability 0.5; otherwise, sample x_n uniformly on points that are within R of $\tilde{x}_{n-1}$ on each dimension, i.e., $\{x \in S : |x^i - \tilde{x}^i_{n-1}| \le R \text{ for all } i = 1, \ldots, d\}$, where x^i is the ith component of x and the radius parameter R is the tuning parameter of the sampler. Given x_n, sample u_n having the same distribution as U and evaluate $y_n = g(x_n, u_n)$.

4.3. ASR with IHR Sampler (IHR-ASR)

Parameters: b and sequence $\{\mathsf{M}(i) = \lfloor i^b \rfloor\}$; c and sequence $\{\mathsf{K}(i) = \lceil i^c \rceil\}$; δ, L, and T.

For each $x \in S$, the algorithm keeps track of the following accumulators: $\mathsf{N}_k(x)$, the number of objective function observations collected at x at iteration k; $\mathsf{S}_k(x)$, the sum of these $\mathsf{N}_k(x)$ objective function observations; and $\hat{f}_k(x) = \mathsf{S}_k(x)/\mathsf{N}_k(x)$.

Step 0: Let $i = 1$, $k = 0$, and $\mathscr{X}_0 = \emptyset$.

Step 1: Let $k \leftarrow k + 1$. If $k = \mathsf{M}(i)$, then go to Step 2. Otherwise, go to Step 3.

Step 2: Sample a new design point by performing the following steps.

Step 2.1: If $i = 1$, sample x_i uniformly on S. Otherwise, sample x_i from the IHR sampler.

Step 2.2: If $i = 1$, accept x_i. Otherwise, accept x_i if $f_{\mathsf{L}}(x_i) \le \hat{f}_{k-1}(x^*_{k-1}) + \delta$, where $f_{\mathsf{L}}(x_i)$ is an average of L observations of $f(x_i)$. If x_i is accepted, then let $\mathscr{X}_k = \mathscr{X}_{k-1} \cup \{x_i\}$. If x_i is rejected, $\mathscr{X}_k = \mathscr{X}_{k-1}$. Update $\mathsf{N}(x_i)$ and $\mathsf{S}(x_i)$.

Step 2.3: For each $x \in \mathcal{X}_k$, if $\mathsf{N}_k(x) < \mathsf{K}(i)$, obtain $\mathsf{K}(i) - \mathsf{N}_k(x)$ additional objective function observations of $f(x)$ and update $\mathsf{N}(x_i)$ and $\mathsf{S}(x_i)$ accordingly.

Step 2.4: Let $i \leftarrow i+1$. Go to Step 4.

Step 3: Resample from the existing design points by performing the following steps. Let k' be the last iteration that a new point is sampled.

Step 3.1: Sample x from $\mathcal{X}_{k-1}$ according to probability $p_k(x)$, where $p_k(x) \propto \exp(\hat{f}_{k'}(x)/T_{k'})$ and $T_{k'} = T/\log(k'+1)$.

Step 3.2: Obtain an estimate of the objective function value at x. Update $\mathsf{N}(x_i)$ and $\mathsf{S}(x_i)$.

Step 3.3: Go to Step 4.

Step 4: Select an estimate of the optimal solution x_k^* from $\mathcal{X}_k^* = \arg\min_{x \in \mathcal{X}_k} \hat{f}_k(x)$.

Step 5: If stopping criterion satisfied, stop. Otherwise, go to Step 1.

4.4. ASR with AP Sampler (AP-ASR)

The same as IHR-ASR but replace Step 2.1 by the AP sampler.

Theorem 4 establishes that IHR-SO and AP-SO satisfy Assumption 4 when the radii of the shrinking balls are chosen appropriately. To satisfy Assumption 4, basically, the algorithm needs to generate enough sample points to fill all the balls and the balls cannot shrink too fast. Sufficient conditions that allow the algorithm to generate enough points are that the sampling densities are bounded away from zero and the feasible region is a convex set. Theorem 4 then specifies an appropriate shrinking rate of the balls.

Theorem 4. *If S is a bounded convex set, r_n is of $\Omega(n^{-\beta})$, where $\beta = (1-\gamma)/d$ and $1/2 < \gamma < 1$, and the algorithm based on SOSA employs sampling densities q_n that are bounded away from zero on S for all n, then the algorithm satisfies Assumption 4.*

Proof. See the e-companion. □

Corollary 3. *If Assumptions 1, 2, and 3 are satisfied and we choose r_n to be $\Omega(n^{-\beta})$, where $\beta = (1-\gamma)/d$, $1/2 < \gamma < 1$, and $i_n \uparrow \infty$ such that $i_n \le n^s$, where $0 < s < \gamma$, then any algorithm based on SOSA with sampling densities q_n that are bounded away from zero on S for all n generates a sequence of optimal value estimates that converges to the optimal value f^* with probability one.*

Proof. By Theorems 3 and 4, Corollary 3 follows. □

Observe that the sampling densities q_n implied by IHR-SO and AP-SO on a compact set are bounded away from zero. Therefore, IHR-SO and AP-SO generate sequences of optimal value estimates that converge to the global optimum with probability one.

From Theorems 3 and 4, the three parameters γ, β, and s of IHR-SO and AP-SO cannot be chosen independently if one would like to guarantee convergence because $\beta = (1-\gamma)/d$ and $0 < s < \gamma$. A value of s, the order of slowing sequence, i_n, identifies how quickly we would like to adopt new optimal value estimates. From pilot experiments, a small value of s will slow down the algorithm. Therefore, we adopt a large value of s close to one, $s = 0.9$. This choice of s dictates a large value of γ, $\gamma = 0.91$, and a small value of β, $\beta = 0.009$ for the 10-dimensional test problems. Fortunately, a small value of β makes the ball shrink slowly and, hence, allows the algorithm to collect more observations of the objective value at each design point, enhancing the accuracy of the objective value estimates.

An appropriate value of the starting ball radius κ_r for IHR-SO and AP-SO is problem dependent. It depends on the size of feasible region, the smoothness of the objective function, and the adaptive sampling strategy q_n. In this study, we experimented with several values for κ_r for each test problem.

The two algorithms based on SOSA (IHR-SO and AP-SO) use the parameter values

$$\gamma = 0.91, \beta = 0.009, s = 0.9, \text{ and } \kappa_r = \begin{cases} 0.1 & \text{for Problem 1,} \\ 1 & \text{for Problem 2.} \end{cases}$$

With this set of parameters, the shrinking ball radii and the slowing sequence are set to

$$r_n = \begin{cases} 0.1n^{-0.009} & \text{for Problem 1,} \\ n^{-0.009} & \text{for Problem 2,} \end{cases} \quad \text{and } i_n = \lfloor n^{0.9} \rfloor.$$

The two algorithms based on ASR (IHR-ASR and AP-ASR) use the same parameter values used in Andradóttir and Prudius (2010),

$$b = 1.1,\ c = 0.5,\ \delta = 0.01,\ \mathsf{L} = 0.1, \text{ and } T = 0.01.$$

The tuning parameter R for AP sampler is set to 0.07 for Problem 1, and 0.4 for Problem 2, for both AP-SO and AP-ASR.

We apply the four algorithms to two problems. The first problem is the shifted sinusoidal problem, from Ali et al. (2005).

Problem 1 (Shifted Sinusoidal Problem).

$$\begin{aligned} \min \quad & \mathbb{E}\left[f(x) + (1 + |f(x)|)U\right] \\ \text{s.t.} \quad & 0 \le x_i \le \pi, \quad i = 1, \ldots, 10, \end{aligned}$$

where

$$f(x) = -\left[2.5\Pi_{i=1}^{10} \sin(x_i - \pi/6) + \Pi_{i=1}^{10} \sin(5(x_i - \pi/6))\right] + 3.5,$$

$x \in \Re^{10}$, and $U \sim$ Uniform$[-0.1, 0.1]$. According to Ali et al. (2005), this problem contains 4,882,813 local optima with a single global optimum at $x^* = (4\pi/6, \ldots, 4\pi/6)$ and $f(x^*) = 0$.

The second problem is the Rosenbrock problem. This problem is also employed as a test case for AP-ASR in the work of Andradóttir and Prudius (2010).

Problem 2 (Rosenbrock Problem).

$$\begin{aligned} &\min \quad \mathbb{E}\big[f(x) + (1 + |f(x)|)U\big] \\ &\text{s.t.} \quad -10 \le x_i \le 10, \quad i = 1, \ldots, 10, \end{aligned}$$

$$\text{where} \quad f(x) = 10^{-6} \times \sum_{i=1}^{d-1} ((1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2),$$

$x \in \Re^{10}$, and $U \sim \text{Uniform}[-0.1, 0.1]$. The global minimum is at $(1, \ldots, 1)$ and $f^* = 0$.

We apply each algorithm to solve each problem 100 times. The initial point for each of these 100 replications is generated according to a uniform distribution on the feasible region and used for each algorithm. Each time we run the algorithms for 12,000 function evaluations. Note that IHR-SO and AP-SO require only one objective function evaluation per iteration. Therefore, each optimization run requires exactly 12,000 iterations. For IHR-ASR and AP-ASR, we count the objective function evaluations as the algorithms progress and stop once 12,000 function evaluations are reached, which may require less than 12,000 iterations. We then record the following four measurements averaged over 100 runs, at each iteration, for $n = 1, 2, \ldots, 12{,}000$:

- the optimal value estimate:

$$\hat{f}_n^* = \min_{x \in \mathcal{X}_n} \hat{f}_n(x),$$

- the (true) objective function of the optimal solution estimate (the best candidate):

$$f(x_n^*), \text{ where } x_n^* \in \arg\min_{x \in \mathcal{X}_n} \hat{f}_n(x),$$

- the contributions to the optimal solution estimate (the counts of the sample points that contribute to the objective function estimate of the optimal solution estimate):

$$l_n^* = |\{k \le n : x_k \in B(x_n^*, r_n)\}|,$$

- the average noise of the optimal solution estimate:

$$\hat{e}_n^* = \frac{\sum_{\{k \le n:\ x_k \in B(x_n^*, r)\}} (1 + |f(x)|)U_k}{l_n^*}.$$

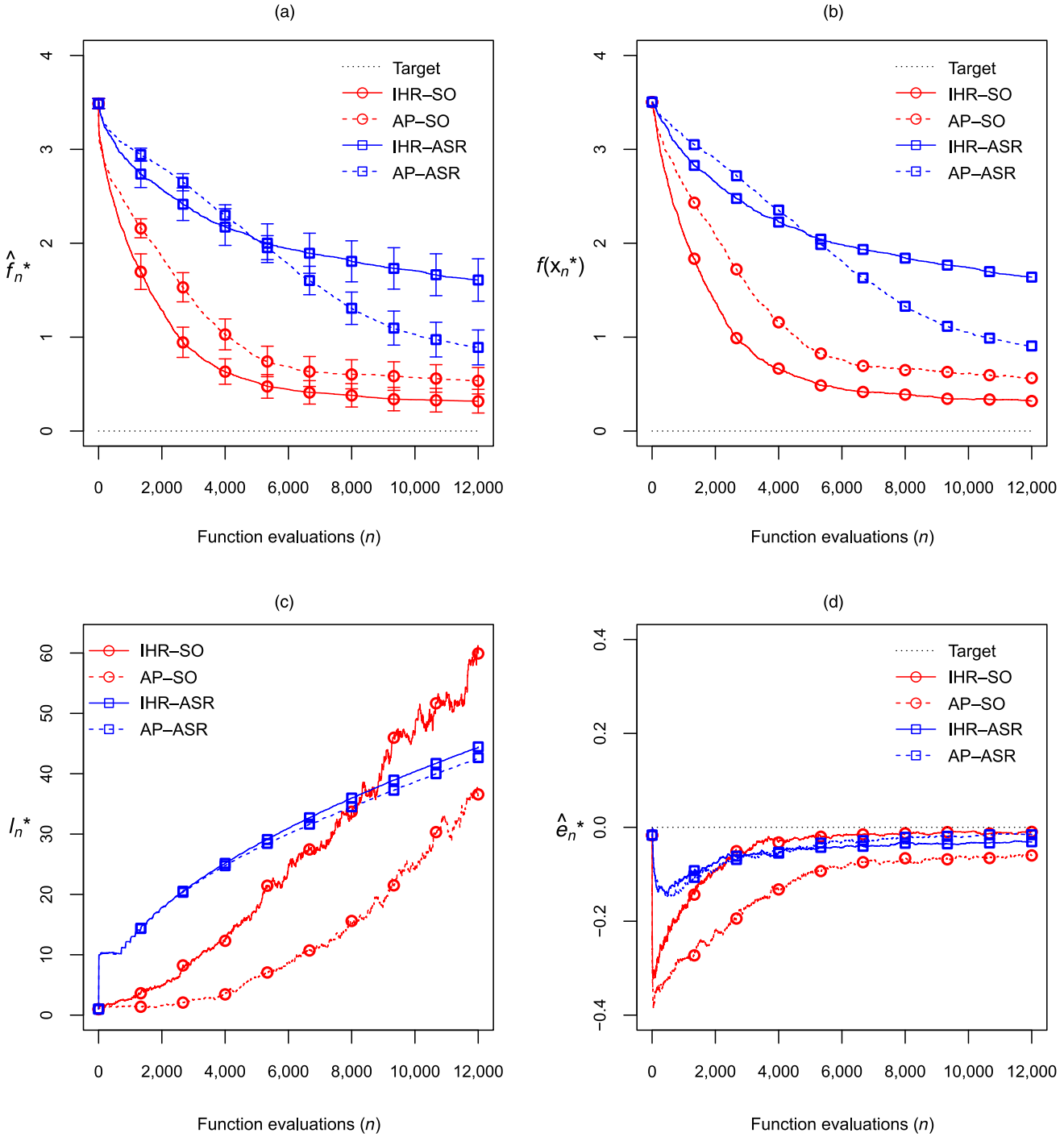
Figure 1 shows the four performance measurements of the four algorithms (IHR-SO, AP-SO, IHR-ASR, and AP-ASR) applied to Problem 1 and averaged over 100 runs at selected number of function evaluations. Panel (a) of Figure 1 also shows the 95% confidence intervals of the optimal value estimates, represented by vertical bars. The confidence interval represents the variation across simulation runs. It is calculated as the mean estimate plus and minus 1.96 times the standard deviation divided by 10 (the square root of the number of simulation runs).

The two SOSA algorithms, IHR-SO and AP-SO, outperform the two ASR algorithms, IHR-ASR and AP-ASR. From panels (a) and (b) of Figure 1, we observe that the optimal value estimates and the objective function value of the optimal solution estimates of SOSA (IHR-SO and AP-SO) converge to the global optimum (target) more quickly than those of ASR (IHR-ASR and AP-SO). Panel (c) of Figure 1 shows how objective function observations accumulate at the optimal solution estimates over the course of each of the algorithms. The errors of the optimal value estimates (panel (d) of Figure 1) decrease as more observations accumulate. The two ASR algorithms are designed to accumulate observations uniformly over the course of the algorithms, as seen from panel (c) of Figure 1. Panel (c) of Figure 1 also shows that IHR-SO accumulates more observations over the course of the run, although adaptively. From panel (c) of Figure 1, IHR-SO and AP-SO are more aggressive at the beginning stage of the algorithms when the quality of the optimal value estimates are poor, i.e., not spending too many observations at the early sample points. The SOSA algorithms then adaptively accumulate more observations for the optimal value estimates in the later stage of the algorithms, when the quality of the estimates get higher. Observe that the ASR algorithms accumulate more observations for the optimal value estimates at the early stage than SOSA algorithms do, slowing the algorithms down. Consequently, as shown in panel (d) of Figure 1, the average noises of the optimal value estimates reduce more rapidly at the early stage in the case of ASR algorithms. Toward the end, the average noises of the four algorithms are not much different. Observe that the errors at the optimal solution estimates are negatively biased. This is a common behavior found in algorithms whose optimal estimates are chosen as the minimum among all the estimates, because the ones that underestimate the true objective function value will be more likely to be chosen, causing the average errors to be negative.

Table 1 shows the statistics of the optimal value estimates $\hat{f}_n^*$ of the four algorithms at termination when applied to Problem 1. As seen from Figure 1, the mean estimates of IHR-SO and AP-SO are closer to the optimal value (zero) than those of IHR-ASR and AP-ASR. For Problem 1, the mean squared errors of IHR-SO and AP-SO are also significantly smaller than those of IHR-ASR and AP-ASR. Furthermore, for Problem 1, IHR-SO consistently outperforms IHR-ASR, and AP-SO outperforms AP-ASR at the 50th and 75th percentiles as well as the worst estimate encountered.

Figure 2 shows the four performance measurements of the four algorithms (IHR-SO, AP-SO, IHR-ASR, and AP-ASR) applied to Problem 2 and averaged over 100

Figure 1. (Color online) Performance Diagnostics for IHR-SO, AP-SO, IHR-ASR, and AP-ASR with Respect to Problem 1



Notes. Panels (a) and (b) exhibit the optimal value estimate (with confidence intervals) and the true objective function value at the optimal solution estimate (the best candidate), respectively. Panels (c) and (d) show the contributions to the best candidates and the average noises of the optimal solution estimate as functions of objective function evaluations, respectively.

runs at each number of function evaluations up to 4,000 function evaluations. Panel (a) of Figure 2 also shows the optimal value estimates with 95% confidence intervals, which are quite tight.

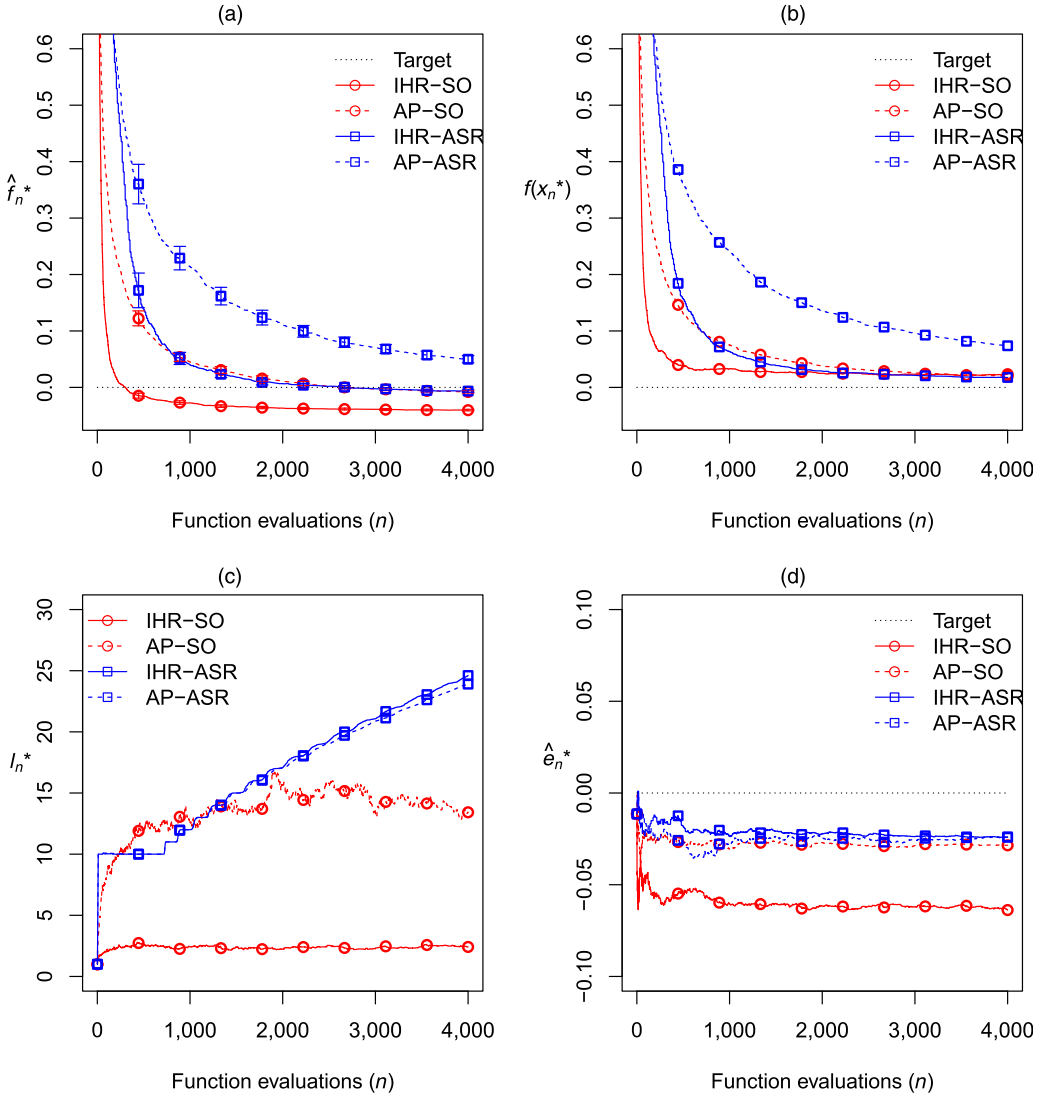
Again, the two SOSA algorithms, IHR-SO and AP-SO, outperform the two ASR algorithms, IHR-ASR and AP-ASR. From panels (a) and (b) of Figure 2, we observe that the optimal value estimates and the objective function value of the optimal solution estimates of SOSA (IHR-SO and AP-SO) converge to the global optimal value more quickly than those of ASR (IHR-ASR and AP-ASR).

Table 1. Statistics of the Optimal Value Estimates $\hat{f}_n^*$ of the Four Algorithms at Termination

					Percentile			
Problem	Algorithm	Mean	Mean squared error	Best	25	50	75	Worst
Problem 1	IHR-SO	0.3181	0.5103	0.0631	0.0915	0.1073	0.1433	2.6569
	AP-SO	0.5354	0.7961	0.1109	0.1211	0.1278	0.9007	2.5117
	IHR-ASR	1.6085	3.8991	0.0751	0.2948	2.5052	2.5897	3.1575
	AP-ASR	0.8905	1.6840	0.0436	0.0922	0.1780	1.8247	2.7173
Problem 2	IHR-SO	−0.0402	0.0017	−0.0524	−0.0466	−0.0407	−0.0352	−0.0247
	AP-SO	−0.0079	0.0002	−0.0231	−0.0145	−0.0104	−0.0025	0.0615
	IHR-ASR	−0.0065	0.0002	−0.0274	−0.0144	−0.0070	−0.0010	0.0243
	AP-ASR	0.0499	0.0035	−0.0020	0.0254	0.0471	0.0682	0.1593

Notes. The experiments for Problem 1 terminate with $n = 12{,}000$. The experiments for Problem 2 terminate with $n = 4{,}000$.

Figure 2. (Color online) Performance Diagnostics for IHR-SO, AP-SO, IHR-ASR, and AP-ASR with Respect to Problem 2



Notes. Panels (a) and (b) exhibit the optimal value estimate (with confidence intervals) and the true objective function value at the optimal solution estimate (the best candidate), respectively. Panels (c) and (d) show the contributions to the best candidates and the average noises of the optimal solution estimate as functions of objective function evaluations, respectively.

Table 1 shows the statistics of the optimal value estimates $\hat{f}_n^*$ of the four algorithms at termination when applied to Problem 2. At termination 4,000, the optimal value estimates of all four algorithms are very close to the true optimal value of zero. The mean squared errors of AP-SO and IHR-ASR are the smallest, followed by that of IHR-SO.

5. Conclusion

We propose a single observation adaptive search (SOSA) framework for continuous simulation optimization. Under this framework, an objective function value estimate of a sample point is formed by the average of the observed function values within a neighborhood of that sample point. The error of an estimate then consists of a bias and a random error. We show that the accumulated random error of each objective function estimate can be decomposed into a nonmartingale fixed term and a progressive martingale term. By slowing down the reporting of the optimal value estimate, the martingale property of the accumulated error ensures that the random error will disappear in the limit. Additionally, the bias is controlled by the shrinking ball mechanism and converges to zero. In conclusion, the optimal value estimate from an algorithm within the SOSA framework converges to the true optimal value with probability one.

We also demonstrate the effectiveness and the efficiency of the framework by modifying two adaptive search algorithms, namely Improving Hit-and-Run (IHR) and Andradóttir–Prudius (AP), to fit this framework. We also show that any algorithm with sampling density bounded away from zero on the feasible region, in particular, IHR-SO and AP-SO, satisfy the four assumptions and, hence, converges with probability one to a global optimum. The two algorithms under the SOSA framework outperform the same two algorithms under the adaptive search with resampling (ASR) alternative, as demonstrated on two noisy objective functions. The performance confirms the theory developed and illustrates the potential of the new continuous simulation optimization paradigm.

Acknowledgments

The authors thank Chulalongkorn University for supporting the coauthors' visits.

References

Ali MM, Khompatraporn C, Zabinsky ZB (2005) A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems. *J. Global Optim.* 31(4):635–672.

Andradóttir S, Prudius AA (2010) Adaptive random search for continuous simulation optimization. *Naval Res. Logist.* 57(6):583–604.

Barton RR, Meckesheimer M (2006) Metamodel-based simulation optimization. *Handbook in Operations Research and Management Science*, vol. 13 (Elsevier, New York), 535–573.

Baumert S, Smith RL (2002) Pure random search for noisy objective functions. Technical Report 01-03, University of Michigan, Ann Arbor.

Borkar VS (2008) *Stochastic Approximation: A Dynamical Systems Viewpoint* (Cambridge University Press, New York).

Borkar VS (2013) Stochastic approximation. *Resonance* 18(12):1086–1094.

Chau M, Fu MC (2015) An overview of stochastic approximation. Fu MC, ed. *Handbook of Simulation Optimization* (Springer, New York), 149–178.

Feller W (1971) *An Introduction to Probability Theory and Its Applications*, vol. 2 (Wiley, New York).

Fu MC, ed. (2015) *Handbook of Simulation Optimization* (Springer, New York).

Ghate A, Smith RL (2008) Adaptive search with stochastic acceptance probabilities for global optimization. *Oper. Res. Lett.* 36(3):285–290.

Homem-De-Mello T (2003) Variable-sample methods for stochastic optimization. *ACM Trans. Model. Comput. Simul.* 13(2):108–133.

Hu J, Fu MC, Marcus SI (2007) A model reference adaptive search algorithm for global optimization. *Oper. Res.* 55:549–568.

Hu J, Zhou E, Fan Q (2014) Model-based annealing random search with stochastic averaging. *ACM Trans. Model. Comput. Simul.* 24(4):1–23.

Kiatsupaibul S, Smith RL, Zabinsky ZB (2015) Improving hit-and-run with single observations for continuous simulation optimization. Yilmaz L, Chan WKV, Moon I, Roeder TMK, Macal C, Rossetti MD, eds. *Proc. 2015 Winter Simulation Conf.* (IEEE, Piscataway, NJ), 3569–3576.

Kim S, Pasupathy R, Henderson SG (2015) A guide to sample average approximation. Fu MC, ed. *Handbook of Simulation Optimization* (Springer, New York), 207–243.

Kleywegt A, Shapiro A, Homem-De-Mello T (2002) The sample average approximation method for stochastic discrete optimization. *SIAM J. Optim.* 12(2):479–502.

Kushner H, Yin G (2003) *Stochastic Approximation and Recursive Algorithms and Applications*, 2nd ed. (Springer, New York).

Linz D, Zabinsky ZB, Kiatsupaibul S, Smith RL (2017) A computational comparison of simulation optimization methods using single observations within a shrinking ball on noisy black-box functions with mixed integer and continuous domains. Chan WKV, D'Ambrogio A, Zacharewicz G, Mustafee N, Wainer G, Page E, eds. *Proc. 2017 Winter Simulation Conf.* (IEEE, Piscataway, NJ), 2045–2056.

Mete HO, Zabinsky ZB (2014) Multi-objective interacting particle algorithm for global optimization. *INFORMS J. Comput.* 26(3): 500–513.

Molvalioglu O, Zabinsky ZB, Kohn W (2009) The interacting-particle algorithm with dynamic heating and cooling. *J. Global Optim.* 43: 329–356.

Molvalioglu O, Zabinsky ZB, Kohn W (2010) Meta-control of an interacting-particle algorithm. *Nonlinear Anal. Hybrid Systems* 4(4):659–671.

Pasupathy R (2010) On choosing parameters in retrospective-approximation algorithms for stochastic root finding and simulation optimization. *Oper. Res.* 58(4-part 1):889–901.

Pasupathy R, Ghosh S (2013) Simulation optimization: A concise overview and implementation guide. Topaloglu H, ed. *Theory Driven by Influential Applications*, TutORials in Operations Research (INFORMS, Catonsville, MD), 122–150.

Pedrielli G, Ng SH (2015) Kriging-based simulation-optimization: A stochastic recursion perspective. Yilmaz L, Chan WKV, Moon I, Roeder TMK, Macal C, Rossetti MD, eds. *Proc. 2015 Winter Simulation Conf.* (IEEE, Piscataway, NJ), 3834–3845.

Robbins H, Monro S (1951) A stochastic approximation method. *Ann. Math. Statist.* 22(3):400–407.

Romeijn HE, Smith RL (1994) Simulated annealing and adaptive search in global optimization. *Probab. Engrg. Inform. Sci.* 8:571–590.

Shapiro A, Dentcheva D, Ruszczyński AP (2009) *Lectures on Stochastic Programming: Modeling and Theory* (Society for Industrial Applied Mathematics, Philadelphia).

Shi L, Ólafsson S (2000a) Nested partitions method for global optimization. *Oper. Res*. 48(3):390–407.

Shi L, Ólafsson S (2000b) Nested partitions method for stochastic optimization. *Methodolology Comput. Appl. Probab*. 2(3):271–291.

Smith RL (1984) Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Oper. Res*. 32(6):1296–1308.

Spall JC (2003) *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control* (Wiley-Interscience, Hoboken, NJ).

Yakowitz S, L'Ecuyer P, Vázquez-Abad F (2000) Global stochastic optimization with low-dispersion point sets. *Oper. Res*. 48(6):939–950.

Zabinsky ZB (2003) *Stochastic Adaptive Search for Global Optimization* (Springer, New York).

Zabinsky ZB (2011) Stochastic search methods for global optimization. Cochran JJ, Cox LA, Keskinocak P, Kharoufeh JP, Smith JC, eds. *Wiley Encyclopedia of Operations Research and Management Science* (John Wiley & Sons, New York).

Zabinsky ZB, Smith RL (2013) Hit-and-run methods. Gass SI, Fu MC, eds. *Encyclopedia of Operations Research & Management Science*, 3rd ed. (Springer, New York).

Zabinsky ZB, Smith RL, McDonald JF, Romeijn HE, Kaufman DE (1993) Improving hit-and-run for global optimization. *J. Global Optim*. 3(2):171–192.

Zhou E, Bhatnagar S, Chen X (2014) Simulation optimization via gradient-based stochastic search. TolkA, Diallo SY, Ryzhov IO, Yilmaz L, Buckley S, Miller JA, eds. *Proc. 2014 Winter Simulation Conf.* (IEEE, Piscataway, NJ), 3869–3879.

Seksan Kiatsupaibul is an associate professor in the Department of Statistics at Chulalongkorn University, Bangkok, Thailand. His current research interests lie at the intersection of optimization and statistics.

Robert L. Smith is the Altarum/ERIM Russell D. O'Neal Professor Emeritus of Engineering at the University of Michigan, Ann Arbor. His current research addresses countably infinite linear programming and simulation optimization. He is an INFORMS Fellow.

Zelda B. Zabinsky is a professor in the Industrial and Systems Engineering Department at the University of Washington, with adjunct appointments in the Departments of Electrical, Mechanical, and Civil and Environmental Engineering. Her research interests include global optimization, simulation optimization, and applications of optimization under uncertainty.