Proceedings of the 1999 Winter Simulation Conference
P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, eds.

STOCHASTIC OPTIMIZATION AND THE SIMULTANEOUS PERTURBATION METHOD

James C. Spall

The Johns Hopkins University
Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723-6099, U.S.A.

ABSTRACT

Multivariate stochastic optimization plays a major role in the analysis and control of many real-world systems. In almost all large-scale practical optimization problems, it is necessary to use a mathematical algorithm that iteratively seeks out the solution because an analytical (closed-form) solution is rarely available. In the above spirit, the "simultaneous perturbation stochastic approximation (SPSA)" method for difficult multivariate optimization problems has been developed. SPSA has recently attracted considerable international attention in areas such as statistical parameter estimation, feedback control, simulation-based optimization, signal and image processing, and experimental design. The essential feature of SPSA—which accounts for its power and relative ease of implementation—is the underlying gradient approximation that requires only two measurements of the objective function regardless of the dimension of the optimization problem. This feature allows for a significant decrease in the cost of optimization, especially in problems with a large number of variables to be optimized.

1 INTRODUCTION

This paper is an introduction to the simultaneous perturbation stochastic approximation (SPSA) algorithm for stochastic optimization of multivariate systems. Optimization algorithms play a critical role in the design, analysis, and control of most physical and nonphysical systems and are in widespread use in the simulation community. Before presenting the SPSA algorithm, we present some general background on the stochastic optimization context of interest here.

The mathematical representation of most optimization problems is the minimization (or maximization) of some scalar-valued objective function with respect to a vector of adjustable parameters. The optimization algorithm is a step-by-step procedure for changing the adjustable parameters from some initial guess (or set of guesses) to a value that offers an improvement in the objective function. Although many optimization algorithms have been developed that assume a deterministic setting and that assume information is available on the gradient of the loss function with respect to the parameters being optimized, our focus here is on the stochastic setting where only measurements of the loss function are available (i.e., no gradient information). (We will, however, present in Section 6 an adaptive method that applies in both the gradient-free and gradient-based cases.) This interest in algorithms without direct gradient information has been motivated, for example, by problems in the adaptive control and statistical identification of complex systems, the optimization of processes by large Monte Carlo simulations, the training of recurrent neural networks, the recovery of images from noisy sensor data, and the design of complex queuing and discrete-event systems. Rather, these algorithms are based on an *approximation* to the gradient formed from (generally noisy) measurements of the loss function.

More specifically, the goal is to minimize a loss function $L(\theta)$ over $\theta \in R^p$, $p \geq 1$. The SPSA algorithm works by iterating from an initial guess of the optimal θ, where the iteration process depends on the above-mentioned "simultaneous perturbation" approximationto the gradient $g(\theta) \equiv \partial L/\partial\theta$. We assume that measurements $y(\theta)$ of the loss function are available at any value of θ:

$$y(\theta) = L(\theta) + \textit{noise}.$$

In some cases, exact loss function measurements will be available; this corresponds to the *noise* = 0 setting. Note that *no* direct measurements (with or without noise) of the gradient $g(\theta)$ are assumed available. This measurement formulation is identical to that of the Kiefer-Wolfowitz finite-difference SA algorithm (Kiefer and Wolfowitz, 1952, and Blum, 1954). In cases where more than one point satisfies $g(\theta) = 0$, then the algorithm may only converge to a local minimum (however, as a consequence of the basic recursive form of the algorithm, there is generally not a risk of converging to a maximum or saddlepoint of $L(\theta)$, i.e., another point where $g(\theta)$ may equal zero). Section 5 will briefly discuss modifications to the basic SPSA algorithm to allow it to search for the global solution among multiple local solutions and to find solutions in the presence of explicit or implicit constraints on feasible θ.

Overall, gradient-free stochastic algorithms exhibit convergence properties similar to the gradient-based stochastic algorithms (e.g., Robbins-Monro 1951, stochastic approximation, i.e., R-M SA) while requiring only loss function measurements. The gradient-based algorithms rely on direct measurements of the gradient of the loss function with respect to the parameters being optimized. These measurements typically yield an *unbiased estimate* of the gradient. A main advantage of the gradient-free algorithms is that they do not require the detailed knowledge of the functional relationship between the parameters being adjusted (optimized) and the loss function being minimized that is required in gradient-based algorithms. Such a relationship can be notoriously difficult to develop in some areas (e.g., simulation-based optimization and nonlinear feedback controller design) while in other areas (such as recursive statistical parameter estimation), there may be large computational savings in calculating a loss function relative to that required in calculating a gradient. On the other hand, in some cases direct gradient observations are used with considerable advantage, including infinitesimal perturbation analysis (IPA) for simulation-based optimization of discrete event systems (Fu and Hu 1997) and backpropagation for neural network training.

Section 2 summarizes the problem setting and describes SPSA and the related finite-difference algorithm. Section 3 discusses some of the theory associated with the convergence and efficiency of SPSA. Section 4 provides a pointer to a step-by-step guide to implementation that is aimed at helping the reader code the algorithm for his or her specific application. Section 5 discusses some extensions to the basic SPSA algorithm. Section 6 summarizes some relatively recent results on a second-order (adaptive) version of SPSA that emulates for stochastic problems the Newton-Raphson algorithm of deterministic optimization. This adaptive SPSA approach applies in either the conventional setting where only loss function measurements are available (i.e., no gradient or Hessian information) or in the stochastic gradient (Robbins-Monro) setting where direct unbiased gradient measurements are available. This paper does not present a numerical study of SPSA as such studies are available in many of the references (see, e.g., Spall 1992; Chin 1997; Fu and Hill 1997; or Spall and Cristion 1994, 1998). Some additional recent applications of SPSA are summarized in Spall (1998a).

2 FINITE DIFFERENCE AND SIMULTANEOUS PERTURBATION STOCHASTIC APPROXIMATION

As motivated above, we now assume that no direct measurements of $\partial L/\partial\theta$ are assumed available (the Robbins-Monro stochastic gradient framework is considered in Section 6). This section will describe the finite-difference SA (FDSA) and simultaneous perturbation SA (SPSA) algorithms. Although the emphasis of this paper is SPSA, the FDSA discussion is included for comparison because FDSA is a classical method for stochastic optimization (Kiefer and Wolfowitz 1952, and Blum 1954).

The SPSA and FDSA procedures are in the general recursive SA form:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \ , \qquad (1)$$

where $\hat{g}_k(\hat{\theta}_k)$ is the estimate of the gradient $g(\theta)$ at the iterate $\hat{\theta}_k$ based on the above-mentioned measurements of the loss function. Under appropriate conditions, the iteration in (1) will converge to θ* in some stochastic sense (usually "almost surely [a.s.]") (see, e.g., Fabian 1971, Kushner and Clark 1978, or Kushner and Yin 1997).

The essential part of (1) is the gradient approximation $\hat{g}_k(\theta_k)$. We discuss below the two forms of interest here. Let c_k be some (usually small) positive number. One-sided gradient approximations involve loss measurements $y(\theta_k)$ and $y(\theta_k + \textit{perturbation})$ while two-sided gradient approximations involve measurements of the form $y(\theta_k \pm \textit{perturbation})$. The two general forms of gradient approximations for use in FDSA and SPSA, respectively, are:

Finite-difference, where each component of $\hat{\theta}_k$ is perturbed one-at-a-time and corresponding measurements $y(\cdot)$ are obtained; each component of the gradient estimate is formed by differencing the corresponding $y(\cdot)$ values and then dividing by a difference interval. This is the standard Kiefer-Wolfowitz approach to approximating gradient vectors and is motivated directly from the definition of a gradient as a vector of p partial derivatives, each constructed as the limit of the ratio of a change in the function value over a corresponding in one component of the argument vector. The two-sided FD approximation is given by

$$\hat{g}_k(\hat{\theta}_k) = \begin{bmatrix} \dfrac{y(\hat{\theta}_k + c_k e_1) - y(\hat{\theta}_k - c_k e_1)}{2c_k} \\ \cdot \\ \cdot \\ \cdot \\ \dfrac{y(\hat{\theta}_k + c_k e_p) - y(\hat{\theta}_k - c_k e_p)}{2c_k} \end{bmatrix} ,$$

where e_i denotes a vector with a one in the i^{th} place and zeros elsewhere (an obvious analogue holds for the one-sided version in Blum, 1954; likewise for the SP form below).

Simultaneous perturbation, has all elements of $\hat{\theta}_k$ randomly perturbed together ("simultaneously") to obtain two measurements $y(\cdot)$, but each component of $\hat{g}_k(\hat{\theta}_k)$ is formed from a ratio involving the difference in the two corresponding measurements and the individual components in the perturbation vector. For two-sided SP, we have

$$\hat{g}_k(\hat{\theta}_k) = \frac{y(\hat{\theta}_k + c_k\Delta_k) - y(\hat{\theta}_k - c_k\Delta_k)}{2c_k}\begin{bmatrix}\Delta_{k1}^{-1}\\ \Delta_{k2}^{-1}\\ \cdot\\ \cdot\\ \cdot\\ \Delta_{kp}^{-1}\end{bmatrix}, \qquad (2)$$

where the distribution of the user-specified p-dimensional random perturbation vector, $\Delta_k = (\Delta_{k1}, \Delta_{k2},\ldots,\Delta_{kp})^T$, satisfies conditions discussed in Section 3 (superscript "T" denotes vector transpose).

Note that the number of loss function measurements $y(\cdot)$ needed in each iteration of FDSA grows with p while with SPSA only *two* measurements are needed *independent* of p since the numerator is the same in all p components. (An approach in the same spirit as SPSA, called random directions SA, is discussed in Polyak and Tsypkin 1973, and Kushner and Clark 1978, but it is shown in Spall 1992, 1998b, and Chin 1997, that SPSA will generally have a lower asymptotic mean-square error than random directions SA for the same number of measurements $y(\cdot)$ since the random directions approach relies on *fourth* moments of its perturbation distribution versus just the second moments for SPSA.) The measurement savings per iteration, of course, provides the *potential* for SPSA to achieve large savings (over FDSA) in the total number of measurements required to estimate θ when p is large. This potential is only realized if the number of iterations required for effective convergence to θ* does not increase in a way to cancel the measurement savings per gradient approximation at each iteration. Section 3 will discuss this efficiency issue further, demonstrating when this potential can be realized by establishing the fundamental result:

Under reasonably general conditions, SPSA and FDSA achieve the same level of statistical accuracy for a given number of iterations although SPSA uses p times fewer function evaluations than FDSA (since each gradient approximation uses only 1/p the number of function evaluations).

3 BASIC ASSUMPTIONS AND SUPPORTING THEORY

Spall (1988, 1992) presents conditions for convergence of the SPSA iterate ($\hat{\theta}_k \rightarrow \theta^*$ a.s.) using the differential equation approach discussed in Ljung (1977) and Kushner and Clark (1978) in the context of the R-M algorithm. Because of the different form of the input, the conditions here are somewhat different from those of the R-M approach. In particular, we must impose conditions on *both* gain sequences (a_k and c_k), the distribution of Δ_k, and the statistical relationship of Δ_k to the measurements $y(\cdot)$ (the R-M algorithm does not, in its basic form, have a c_k sequence). These conditions ensure convergence of $\hat{\theta}_k$ to a minimizing point θ*. Alternative conditions for convergence of SPSA have been presented in Wang and Chong (1996), Dippon and Renz (1997) (Dippon and Renz also present an asymptotic normality result under conditions slightly different from those given below), Chen, et al. (1999), and Gerencsér (1999).

Although the convergence result for SPSA is of some independent interest, the most interesting theoretical results in Spall (1992), and those that most justify the use of SPSA, are the asymptotic efficiency conclusions that follow from an asymptotic normality result. This result follows from conditions given in Spall (1992) showing that

$$k^{\beta/2}(\hat{\theta}_k - \theta^*) \xrightarrow{\text{dist.}} N(\mu, \Sigma) \text{ as } k \rightarrow \infty, \qquad (3)$$

where $\xrightarrow{\text{dist.}}$ denotes convergence in distribution, $\beta > 0$, μ and Σ are a mean vector and covariance matrix. Here, μ depends on both the Hessian and the third derivatives of $L(\theta)$ at θ* and Σ depends on the Hessian matrix at θ*. (Note that in general, $\mu \neq 0$ is in contrast to many well-known asymptotic normality results in estimation, including those for the R-M algorithm. As with the R-M case, the asymptotic distribution result (3) allows one to determine optimal gain decay rates, i.e., rates that provide the maximum value of β/2. This maximum value is $\beta/2 = 1/3$. Hence the fastest possible rate at which the error $\hat{\theta}_k - \theta^*$ will go to zero is $k^{-1/3}$ (so that the quantity on the left-hand side of (3) is properly "balanced" when considering the $k^{\beta/2} = k^{1/3}$ multiplier). This contrasts with the fastest allowable rate of $k^{-1/2}$ for the R-M SA algorithm. Hence, one measure of the value of the gradient information in R-M is the increase in rate of convergence. (Kleinman, et al., 1999 discusses one case where it also possible to get a $k^{-1/2}$ rate of convergence in SPSA through the use of common random numbers in a simulation-based optimization context.)

Spall (1992, Sect. 4) uses the asymptotic normality result in expression (3) (together with a parallel result for

FDSA) to establish the relative efficiency of SPSA. This efficiency depends on the shape of $L(\theta)$, the values for $\{a_k\}$ and $\{c_k\}$, and the distributions of the $\{\Delta_{ki}\}$ and measurement noise terms $\{\varepsilon_k^{(\pm)}\}$. There is no single expression that can be used to characterize the relative efficiency. However, as discussed in Spall (1992, Sect. 4) and Chin (1997), in most practical problems, SPSA will be asymptotically more efficient than FDSA. In particular, by equating the asymptotic mean-squared errors $E\left(\left\|\hat{\theta}_k - \theta^*\right\|^2\right)$ in SPSA and FDSA, we find

$$\frac{\text{no. of } y(\theta) \text{ values in SPSA}}{\text{no. of } y(\theta) \text{ values in FDSA}} \to \frac{1}{p} \qquad (4)$$

as the number of loss measurements in both procedures gets large. Hence, expression (4) implies that the p-fold saving per iteration (gradient approximation) translates directly into a p-fold savings in the overall optimization process.

4 PRACTICAL IMPLEMENTATION OF SPSA

Spall (1998c) provides a convenient step-by-step summary for implementation of SPSA. This summary includes simple MATLAB code for implementing the steps. Also included are practical guidelines in choosing the gain sequences $a_k = a/(A+k+1)^\alpha$ and $c_k = c/(k+1)^\gamma$ with a, c, A, α, and γ being non-negative constants.

5 OPTIMAL PERTURBATION DISTRIBUTION; APPLICATIONS IN CONTROL SYSTEMS; GLOBAL, DISCRETE, AND CONSTRAINED OPTIMIZATION

Sadegh and Spall (1998) consider the problem of choosing the best distribution for the Δ_k vector. Based on asymptotic distribution results, it is shown that the optimal distribution for the components of Δ_k is symmetric Bernoulli. This simple distribution has also proven effective in many finite-sample practical and simulation examples. The recommendation in the algorithm description mentioned in Section 4 follows from these findings. It should be noted, however, that other distributions are sometimes desirable. Since the user has full control over this choice and since the generation of Δ_k represents a trivial cost towards the optimization, it may be worth evaluating other possibilities in some applications. For example, Maeda and De Figueiredo (1997) used a symmetric two-part uniform distribution, i.e., a uniform distribution with a section removed near 0 (to preserve the finiteness of inverse moments), in an application for robot control.

Some extensions to the basic SPSA algorithm above are reported in the literature. For example, its use in feedback control problems, where the loss function changes with time, is given in Spall and Cristion (1994, 1998). The 1998 reference is the most complete methodological and theoretical treatment. A *one*-measurement form of the SP gradient approximation is considered in Spall (1997). Although it is shown in this reference that the standard two-measurement form will usually be more efficient (in terms of total number of loss function measurements to obtain a given level of accuracy in the θ iterate), there may be advantages to the one-measurement form in real-time operations, such as feedback control, where the underlying system dynamics may change too rapidly to get a credible gradient estimate with two successive measurements. The Spall and Cristion (1994) reference also reports on a gradient smoothing idea (analogous to “momentum” in the neural network literature) that may help reduce noise effects and enhance convergence (and also gives guidelines for how the smoothing should be reduced over time to ensure convergence). Alternatively, it is possible to average several SP gradient approximations at each iteration to reduce noise effects (at the cost of additional function measurements); this is discussed in Spall (1992).

Implementations of SPSA for *global* minimization are discussed in Chin (1994) and Maryak and Chin (1999). The Chin approach is based on a step-wise (slowly decaying) sequence c_k (and possibly a_k) and the Maryak/Chin approach is based on the principle of injected Monte Carlo noise in the right-hand side of the basic SPSA updating step. This latter approach is a common way of converting SA algorithms to global optimizers (e.g., Yin, 1999). Discrete optimization problems (where θ may take on discrete or combined discrete/continuous values) are discussed in Gerencsér, et al. (1999). The discrete SPSA relies on a fixed-gain (constant a_k and c_k) version of the standard SPSA method. The loss function is assumed to be convex and, in the process of optimization, is temporarily extended to a unique, continuous convex function. The (unique) continuous extension is then used to form a gradient approximation, which is used in a fixed-gain SA algorithm. The parameter estimates produced are constrained to lie on the discrete-valued grid. The problem of constrained (equality and inequality) optimization with SPSA is considered in Sadegh (1997) and Fu and Hill (1997) using a projection approach. While the projection approach has an elegant mathematical form and may be easy to implement, it is quite restricted in the types of constraints that can be practically handled. (Essentially, the constraints must be represented explicitly in θ in a “nice” way so as to facilitate the mapping of a constraint violation in θ to the nearest valid point. In practice this usually boils down to simple componentwise constraints stating that individual components of θ are bounded below and above by particular constants.) An alternative approach to constrained optimization is given in Wang and Spall (1999). This approach is based on altering the loss function to include a penalty term. In particular, at iteration k, $L(\theta)$ is replaced by a modified loss function

$$L(\theta) + r_k P(\theta) ,$$

where r_k is an increasing sequence of positive scalars ($r_k \rightarrow \infty$) and $P(\theta)$ is a penalty function that takes on (usually large) positive values when the constraints are violated. In many practical problems constraints are only implicit in θ, and the penalty function approach is well-suited to handle such cases (e.g., if it is required that $0 \leq h(\theta) \leq 1$ for some function $h(\cdot)$, then $P(\theta)$ can be chosen to penalize values of θ such that $h(\theta)$ is outside of [0, 1] without specifying any explicit constraints on the components of θ).

6 ADAPTIVE SIMULTANEOUS PERTURBATION APPROACH

6.1 Introduction and Basic Algorithm

Based on the simultaneous perturbation idea, this section presents a general adaptive SA algorithm that is based on a simple method for estimating the Hessian matrix while concurrently estimating the primary parameters of interest (θ). This adaptive approach produces a stochastic analogue to the deterministic Newton-Raphson algorithm, hence producing a recursion that is optimal or near-optimal in its rate of convergence and asymptotic error (see Subsection 6.3). The approach applies in both the gradient-free setting stressed above and in the root-finding/stochastic gradient-based (Robbins-Monro) setting. Like the standard (first-order) SPSA algorithm, the algorithm requires only a small number of loss function (or gradient, if relevant) measurements per iteration—independent of the problem dimension—to adaptively estimate the Hessian and parameters of primary interest.

Before the approach is presented, it is useful to contrast it with other second-order SA approaches. A more complete discussion on related work is given in Spall (1999). In the gradient-free setting, Fabian (1971) forms estimates of the gradient and Hessian for a Newton-Raphson-type SA algorithm by using, respectively, a finite-difference approximation and a set of differences of finite-difference approximations. This leads to $O(p^2)$ measurements of $L(\cdot)$ per update of the θ estimate, which is extremely costly when p is large. Ruppert (1985) assumes that direct measurements of the gradient $g(\cdot)$ are available, as in the Robbins-Monro algorithm. He then forms a Hessian estimate by taking a finite difference of gradient measurements; hence, $O(p)$ measurements of $g(\cdot)$ are required for each update step in estimating θ. A type of second-order optimal convergence for SA is reported in Ruppert (1991) and Polyak and Juditsky (1992) based on the idea of iterate averaging. However, as discussed in Maryak (1997), Dippon and Renz (1997), and Spall (1999), this approach may not provide optimal asymptotic convergence in the gradient-free setting and may perform relatively poorly in practical finite-sample problems. The algorithm here is in the spirit of adaptive (matrix) gain SA algorithms such as those considered in Benveniste et al. (1990, Chaps. 3–4) in that a matrix gain is estimated concurrently with an estimate of the parameters of interest. It differs, however, in the relative lack of prior information required (especially in the gradient-free case) and in the small number of loss and/or gradient measurements needed per iteration.

The second-order ASP approach is composed of two parallel recursions: one for θ and one for the Hessian of $L(\theta)$. The two core recursions are, respectively,

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \overline{\overline{H}}_k^{-1} G_k(\hat{\theta}_k), \quad \overline{\overline{H}}_k = f_k(\overline{H}_k) , \qquad \text{(5a)}$$

$$\overline{H}_k = \frac{k}{k+1}\overline{H}_{k-1} + \frac{1}{k+1}\hat{H}_k , \; k = 0, 1, 2, \qquad \text{(5b)}$$

where a_k is a non-negative scalar gain coefficient, $G_k(\hat{\theta}_k)$ is the input information related to $g(\hat{\theta}_k)$ (i.e., the gradient approximation from $y(\cdot)$ measurements in the gradient-free case or the direct observation as in the Robbins-Monro gradient-based case), f_k: $\boldsymbol{R}^{p\times p} \longrightarrow$ {*Positive definite* $p \times p$ *matrices*} is a mapping designed to cope with possible nonpositive-definiteness of $\overline{H}_k$, and $\hat{H}_k$ is a per-iteration estimate of the Hessian discussed below. Eqn. (5a) is a stochastic analogue of the well-known Newton-Raphson algorithm of deterministic search and optimization. Eqn. (5b) is simply a recursive calculation of the sample mean of the per-iteration Hessian estimates. Initialization of the two recursions is discussed in Subsection 6.2 below. Since $G_k(\hat{\theta}_k)$ has a known form, the parallel recursions in eqns. (5a,b) can be implemented once $\hat{H}_k$ is specified. The remainder of this section will focus on two specific implementations of the ASP approach above: 2SPSA (2nd-order SPSA) for applications in the gradient-free case and 2SG (2nd-order stochastic gradient) for applications in the Robbins-Monro gradient-based case.

We now present the per-iteration Hessian estimate $\hat{H}_k$. As with the "basic" first-order SPSA algorithm, let c_k be a positive scalar (decaying to 0 for formal convergence) and $\Delta_k \in R^p$ be a user-generated mean-zero random vector; conditions on c_k, Δ_k, and other relevant quantities are given in Spall (1999). These conditions are close to those of basic

SPSA (e.g., Δ_k being a vector of independent Bernoulli ±1 random variables satisfies these conditions, but a vector of uniformly or normally distributed random variables does not). Examples of valid gain sequences are given below. It will prove convenient to work with a "vector-divide" operation where the ij^{th} element of the resulting matrix corresponds to the ratio of the j^{th} element of the numerator row vector to the i^{th} element of the denominator column vector. Applying the vector-divide operator, the formula for estimating the Hessian at each iteration is:

$$\hat{H}_k = \tfrac{1}{2}\left[\frac{\delta G_k^T}{2c_k\Delta_k} + \left(\frac{\delta G_k^T}{2c_k\Delta_k}\right)^T\right], \tag{6}$$

where

$$\delta G_k = G_k^{(1)}(\hat{\theta}_k + c_k\Delta_k) - G_k^{(1)}(\hat{\theta}_k - c_k\Delta_k)\ ,$$

and $G_k^{(1)}(\cdot)$ may or may not equal $G_k(\cdot)$ depending on the setting. In particular, for 2SPSA, there are advantages to using a *one-sided* gradient approximation in order to reduce the total number of function evaluations (vs. the two-sided form usually recommended for $G_k(\cdot)$) while for 2SG, usually $G_k^{(1)}(\cdot) = G_k(\cdot)$. Note that all elements of $\hat{\theta}_k$ are varied simultaneously (and randomly) in forming $\hat{H}_k$, as opposed to the finite-difference forms in, e.g., Fabian (1971) and Ruppert (1985), where the elements of θ are changed deterministically one at a time. The symmetrizing operation in (6) is convenient in the optimization case being emphasized here to maintain a symmetric Hessian estimate in finite samples. In the general root-finding case, where $H(\theta)$ represents a Jacobian matrix, the symmetrizing operation should not be used since the Jacobian is not necessarily symmetric.

While the ASP structure in (5a,b) and (6) is general, we will largely restrict ourselves in our choice of $G_k(\cdot)$ (and $G_k^{(1)}(\cdot)$) in the remainder of the discussion in order to present concrete theoretical and numerical results. For 2SPSA, we will consider the simultaneous perturbation approach for generating $G_k(\cdot)$ and $G_k^{(1)}(\cdot)$ while for 2SG we will suppose that $G_k(\cdot) = G_k^{(1)}(\cdot)$ is an unbiased direct measurement of $g(\cdot)$ (i.e., $G_k(\cdot) = G_k^{(1)}(\cdot) = g(\cdot)$ + *mean-zero noise*). The rationale for basic SPSA in the gradient-free case was discussed above. In the gradient-based case, SG methods include as special cases the well-known approaches mentioned in Section 1 (backpropagation, etc.). SG methods are themselves special cases of the general Robbins-Monro root-finding framework and, in fact, most of the results here can apply in this root-finding setting as well.

For 2SPSA, the core gradient approximation $G_k(\hat{\theta}_k)$ is taken as $\hat{g}_k(\hat{\theta}_k)$ in eqn. (2), requiring two measurements of $L(\cdot)$, $y(\hat{\theta}_k + c_k\Delta_k)$ and $y(\hat{\theta}_k - c_k\Delta_k)$. In addition to this gradient approximation, these two measurements are employed toward generating the one-sided gradient approximations $G_k^{(1)}(\hat{\theta}_k \pm c_k\Delta_k)$ used in forming $\hat{H}_k$. Two additional measurements $y(\hat{\theta}_k \pm c_k\Delta_k + \tilde{c}_k\tilde{\Delta}_k)$ are used in generating the one-sided approximations as follows:

$$G_k^{(1)}(\hat{\theta}_k \pm c_k\Delta_k) = \frac{y(\hat{\theta}_k \pm c_k\Delta_k + \tilde{c}_k\tilde{\Delta}_k) - y(\hat{\theta}_k \pm c_k\Delta_k)}{\tilde{c}_k}\begin{bmatrix}\tilde{\Delta}_{k1}^{-1}\\ \tilde{\Delta}_{k2}^{-1}\\ \cdot\\ \cdot\\ \cdot\\ \tilde{\Delta}_{kp}^{-1}\end{bmatrix}, \tag{7}$$

with $\tilde{\Delta}_k = (\tilde{\Delta}_{k1}, \tilde{\Delta}_{k2}, \ldots., \tilde{\Delta}_{kp})^T$ generated in the same statistical manner as Δ_k, but independently of Δ_k (in particular, choosing $\tilde{\Delta}_{ki}$ as independent Bernoulli ±1 random variables is a valid—but not necessary—choice), and with $\tilde{c}_k$ satisfying conditions similar to c_k (although the numerical value of $\tilde{c}_k$ may be best chosen larger than c_k; see Subsection 6.2).

Let us summarize some examples of gains that satisfy the conditions in Spall (1999) for convergence and asymptotic normality of 2SPSA and 2SG. For both implementations, we can take a_k and c_k in the form given in Section 3 (condition (vi)). For 2SPSA, we also have $\tilde{c}_k = \tilde{c}/(k+1)^\gamma$, $\tilde{c} > 0$. With these gain forms, examples of specific coefficient values for 2SPSA are: $\alpha = 0.602$, $\gamma = 0.101$ or $\alpha = 1$, $\gamma = 1/6$. For 2SG, $½ < \alpha \le 1$ is valid together with $0 < \gamma < ½$.

Some discussion is included in Spall (1999) providing some informal motivation for the $\hat{H}_k$ form in eqn. (6). This discussion shows that the Hessian estimate is an unbiased estimate to within $O(c_k^2)$. This reference also includes some more rigorous discussion on the form of the estimate and its effect on the convergence properties of ASP.

6.2 Implementation Aspects of ASP

The two recursions in (5a,b) are the foundation for the ASP approach. However, as is typical in all stochastic algorithms, the specific implementation details are important. Eqns. (5a,b) do not fully define these details. Five useful guidelines are given in Spall (1999).

6.3 Theory on Convergence and Efficiency of ASP

Spall (1999) presents some asymptotic theory showing the a.s. convergence of $\hat{\theta}_k$ and $\overline{H}_k$ to θ* and H(θ*), respectively, in both the 2SPSA and 2SG settings. Further, conditions are shown for asymptotic normality of an appropriately normalized estimate $\hat{\theta}_k$. The asymptotic normality results are then used to analyze the asymptotic efficiency of the general ASP approach. To summarize these asymptotic efficiency results, let RMS^*_{SPSA} and RMS^*_{SG} represent the *best possible* root-mean square error of the normalized $\hat{\theta}_k$ when using the SPSA (gradient-free) and SG (gradient-based) approach. These require a choice of gain sequences that use, respectively, exact information on the third and second derivatives of $L(\theta)$ (Dippon and Renz, 1997). This information, of course, is generally unavailable. Hence, RMS^*_{SPSA} and RMS^*_{SG} represent ideal values that will usually be unobtainable in practice. Letting RMS_{2SPSA} and RMS_{2SG} denote the limiting RMS errors for the normalized 2SPSA and 2SG estimates when $a_k = 1/(k + 1)$, we find:

$$\frac{RMS_{2SPSA}}{RMS^*_{SPSA}} < 2 \quad \text{and} \quad \frac{RMS_{2SG}}{RMS^*_{SG}} = 1 . \qquad (8)$$

The interpretation of (8) is that for the SPSA setting, the 2SPSA algorithm can produce an estimate that has an asymptotic RMS error no more than twice the error possible from the best possible (infeasible) algorithm. For the SG setting, the 2SG algorithm produces an error that is asymptotically equal to the best possible. Numerical studies in Spall (1999) show the power of the ASP (2SPSA and 2SG) approach. Luman (1999) applies 2SPSA in a simulation-based optimization approach and demonstrates the improvement possible over basic SPSA when there is very different scaling of the elements in θ (i.e., an illustration of the value of the above-mentioned transform invariance property).

7 CONCLUDING REMARKS

Relative to standard deterministic methods, stochastic optimization considerably broadens the range of practical problems for which one can find rigorous optimal solutions. Algorithms of the stochastic optimization type allow for the effective treatment of problems in areas such as network analysis, simulation-based optimization, pattern recognition and classification, neural network training, image processing, and nonlinear control. It is expected that the role of stochastic optimization will continue to grow as modern systems increase in complexity and as population growth and dwindling natural resources force trade-offs that were previously unnecessary.

The SPSA algorithm has proven to be an effective stochastic optimization method. Its primary virtues are (i) relative ease of implementation and lack of need for loss function gradient, (ii) theoretical and experimental support for relative efficiency, (iii) robustness to noise in the loss measurements, and (iv) empirical evidence of ability to find a global minimum when multiple (local and global) minima exist. Except as discussed in Section 5, SPSA is primarily limited to continuous-variable problems. Numerical comparisons with techniques such as the (Kiefer-Wolfowitz) finite-difference method, simulated annealing, genetic algorithms, and random search have supported the claims of SPSA's effectiveness in a wide range of practical problems. Theoretical evidence also supports the relative efficiency of SPSA in comparison to other popular approaches (Spall, et al. 1999). The rapidly growing number of applications throughout the world provide further evidence of the algorithm's effectiveness. To add to the effectiveness, there have been some extensions of the basic idea, including a stochastic analogue of the fast deterministic Newton-Raphson (second-order) algorithm, adaptations for real-time (control) implementations, and versions for some types of constrained, global, and discrete optimization problems. While much work continues in extending the basic algorithm to a broader range of real-world settings, SPSA addresses a wide range of difficult problems and should likely be considered for many of the stochastic optimization challenges encountered in practice.

REFERENCES

Benveniste, A., Metivier, M., and Priouret, P. (1990), *Adaptive Algorithms and Stochastic Approximations*, Springer-Verlag, New York.

Blum, J.R. (1954), "Multidimensional Stochastic Approximation Methods," *Annals of Mathematical Statistics*, vol. 25, pp. 737-744.

Chen, H.F., Duncan, T.E., and Pasik-Duncan, B. (1999), "A Kiefer-Wolfowitz Algorithm with Random Differences," *IEEE Transactions on Automatic Control*, vol. 44, pp. 442-453.

Chin, D. C. (1994), "A More Efficient Global Optimization Algorithm Based on Styblinski and Tang," *Neural Networks*, vol. 7, pp. 573-574.

Chin, D.C. (1997), "Comparative Study of Stochastic Algorithms for System Optimization Based on Gradient Approximations," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 27, pp. 244-249.

Dippon, J. and Renz, J. (1997), "Weighted Means in Stochastic Approximation of Minima," *SIAM Journal of Control and Optimization*, vol. 35, pp. 1811-1827.

Fabian, V. (1971), "Stochastic Approximation," *Optimizing Methods in Statistics* (Rustagi, J. J., ed.), Academic Press, New York, pp. 439-470.

Fu, M.C. and Hill, S.D. (1997), "Optimization of Discrete Event Systems via Simultaneous Perturbation Stochastic Approximation," *Transactions of the Institute of Industrial Engineers*, vol. 29, pp. 233-243.

Fu, M. C. and Hu, J.-Q. (1997), *Conditional Monte Carlo: Gradient Estimation and Optimization Applications*, Kluwer, New York.

Gerencsér, L. (1999), "Convergence Rate of Moments in Stochastic Approximation with Simultaneous Perturbation Gradient Approximation and Resetting," *IEEE Transactions on Automatic Control*, vol. 44, pp. 894-905.

Gerencsér, L, Hill, S.D., and Vágó, Z. (1999), "Fixed Gain SPSA for Discrete Optimization," *Proceedings of the IEEE Conference on Decision and Control*, to appear.

Kiefer, J. and Wolfowitz, J. (1952), "Stochastic Estimation of a Regression Function," *Annals of Mathematical Statistics*, vol. 23, pp. 462-466.

Kleinman, N.L., Spall, J.C., and Naiman, D.Q. (1999), "Simulation-Based Optimization with Stochastic Approximation Using Common Random Numbers," *Management Science*, to appear.

Kushner, H. J. and Clark, D. S. (1978), *Stochastic Approximation for Constrained and Unconstrained Systems*, Springer-Verlag, New York.

Kushner, H. J. and Yin, G. G. (1997), *Stochastic Approximation Algorithms and Applications*, Springer-Verlag, New York.

Ljung, L. (1977), "Analysis of Recursive Stochastic Algorithms," *IEEE Transactions on Automatic Control*, vol. AC-22, pp. 551-575.

Luman, R.R. (1999), "Upgrading Complex Systems of Systems: A CAIV Methodology for Warfare Area Requirements Analysis," *Military Operations Research*, to appear.

Maeda, Y. and De Figueiredo, R.J. P (1997), "Learning Rules for Neuro-Controller via Simultaneous Perturbation," *IEEE Transactions on Neural Networks*, vol. 8, pp. 1119-1130.

Maryak, J.L. (1997), "Some Guidelines for Using Iterate Averaging in Stochastic Approximation," *Proceedings of the IEEE Conference on Decision and Control*, pp. 2287-2290.

Maryak, J.L. and Chin, D.C. (1999), "Efficient Global Optimization Using SPSA," *Proceedings of the American Control Conference*, pp. 890-894.

Polyak, B. T. and Juditsky, A. B. (1992), "Acceleration of Stochastic Approximation by Averaging," *SIAM Journal on Control and Optimization*, vol. 30, pp. 838-855.

Polyak, B.T. and Tsypkin, Y.Z. (1973), "Pseudogradient Adaptation and Training Algorithms," *Automation and Remote Control*, vol. 34, pp. 377-397.

Robbins, H. and Monro, S. (1951), "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, vol. 22, pp. 400-407.

Ruppert, D. (1985), "A Newton-Raphson Version of the Multivariate Robbins-Monro Procedure," *Annals of Statistics*, vol. 13, pp. 236-245.

Ruppert, D. (1991), "Stochastic Approximation," *Handbook of Sequential Analysis*, (Ghosh, B.K. and Sen, P.K., eds.), Marcel Dekker, New York, pp. 503-529.

Sadegh, P. (1997), "Constrained Optimization via Stochastic Approximation with a Simultaneous Perturbation Gradient Approximation," *Automatica*, vol. 33, pp. 889-892.

Sadegh, P. and Spall, J. C. (1998), "Optimal Random Perturbations for Stochastic Approximation with a Simultaneous Perturbation Gradient Approximation," *IEEE Transactions on Automatic Control*, vol. 43, pp. 1480-1484 (correction to references: vol. 44, p. 231).

Spall, J.C. (1988), "A Stochastic Approximation Algorithm for Large-Dimensional Systems in the Kiefer-Wolfowitz Setting," *Proceedings of the IEEE Conference on Decision and Control*, pp. 1544-1548.

Spall, J.C. (1992), "Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation," *IEEE Transactions on Automatic Control*, vol. 37, pp. 332-341.

Spall, J.C. (1997), "A One-Measurement Form of Simultaneous Perturbation Stochastic Approximation," *Automatica*, vol. 33, pp. 109-112.

Spall, J. C. (1998a), "An Overview of the Simultaneous Perturbation Method for Efficient Optimization," *Johns Hopkins APL Technical Digest*, vol. 19, pp. 482-492.

Spall, J. C. (1998b), Review of Book "Stochastic Approximation Algorithms and Applications" (by H.J. Kushner and G.G. Yin), *IEEE Transactions on Automatic Control*, vol. 43, pp. 753-755.

Spall, J.C. (1998c), "Implementation of the Simultaneous Perturbation Algorithm for Stochastic Optimization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 817-823.

Spall, J.C. (1999), "Adaptive Stochastic Approximation by the Simultaneous Perturbation Method," *IEEE Transactions on Automatic Control*, to appear (in condensed form in *Proceedings of the IEEE Conference on Decision and Control*, 1998, pp. 3872-3879).

Spall, J.C. and Cristion, J.A. (1994), "Nonlinear Adaptive Control Using Neural Networks: Estimation Based on a Smoothed Form of Simultaneous Perturbation

Gradient Approximation," *Statistica Sinica*, vol. 4, pp. 1-27.

Spall, J.C. and Cristion, J.A. (1998), "Model-Free Control of Nonlinear Stochastic Systems with Discrete-Time Measurements," *IEEE Transactions on Automatic Control*, vol. 43, pp. 1198-1210.

Spall, J.C., Hill, S.D., and Stark, D.S. (1999), "Some Theoretical Comparisons of Evolutionary Computation and Other Optimization Approaches," *Proceedings of the Congress on Evolutionary Computation*, 6-9 July, Washington, DC, pp. 1398-1405.

Wang, I.-J. and Chong, E.K.P. (1996), "A Deterministic Analysis of Simultaneous Perturbation Stochastic Approximation," *Proceedings of the 30th Conference on Information Sciences and Systems*, pp. 918-922.

Wang, I.-J. and Spall, J.C. (1999), "A Constrained Simultaneous Perturbation Stochastic Approximation Algorithm Based on Penalty Functions," *Proceedings of the American Control Conference*, pp. 393-399.

Yin, G. (1999), "Rates of Convergence for a Class of Global Stochastic Optimization Algorithms," *SIAM Journal on Optimization*, to appear.

AUTHOR BIOGRAPHY

JAMES C. SPALL joined The Johns Hopkins University, Applied Physics Laboratory in 1983 and was appointed to the Principal Professional Staff in 1991. He also teaches in the Johns Hopkins Part-Time Engineering Division and is Chairman of its Applied Mathematics Program. Dr. Spall has published many articles in the areas of statistics and control and holds two U.S. patents. For the year 1990, he received the Hart Prize as principal investigator of an outstanding Independent Research and Development project at JHU/APL, and in 1997 he was the Chairman of the 4th Symposium on Research and Development at APL. He is an Associate Editor for the *IEEE Transactions on Automatic Control*, a Contributing Editor for the *Current Index to Statistics*, editor and coauthor for the book *Bayesian Analysis of Time Series and Dynamic Models*, and author of the forthcoming textbook *Introduction to Stochastic Search and Optimization* (Wiley). Dr. Spall is a senior member of IEEE, a member of the American Statistical Association, and a fellow of the engineering honor society Tau Beta Pi.