Proceedings of the 38th
Conference on Decision & Control
Phoenix, Arizona USA • December 1999

OPTIMIZATION OVER DISCRETE SETS VIA SPSA

László Gerencsér
Computer and Automation Institute of
the Hungarian Academy of Sciences,
13-17 Kende u., Budapest, 1111, Hungary
gerencser@sztaki.hu

Stacy D. Hill
Applied Physics Laboratory
John Hopkins University
John Hopkins Rd.
Laurel, MD 20723-6099
HillSD1@central.SSD.JHUAPL.edu

Zsuzsanna Vágó
Computer and Automation Institute of
the Hungarian Academy of Sciences,
13-17 Kende u., Budapest, 1111, Hungary
vago@oplab.sztaki.hu

Abstract

A fixed gain version of the SPSA (simultaneous perturbation stochastic approximation) method for function minimization is developed and the error process is characterized. The new procedure is applicable to optimization problems over $\mathbb{Z}^p$, the grid of points in $\mathbf{R}^p$ with integer components. Simulation results and a closely related application, a resource allocation problem, is shortly described.

Keywords. Fixed gain stochastic approximation; higher order difference schemes; ODE-method; resource allocation; asynchronous stochastic approximation.

1 Introduction

The simultaneous perturbation stochastic approximation (SPSA) method developed in [12] is considered an efficient tool for the solution of difficult optimization problems. It is essentially a randomized Kiefer-Wolfowitz method where the gradient is estimated using only two measurements per iteration. The method is particularly suited to problems where the cost function can be computed only by expensive simulations (cf. [2]). The almost sure convergence, the limit distribution and the rate of convergence of higher order moments of the estimator process have been established or reported in a series of papers [4], [9], [8] [12].

The main objective of this paper is to develop an appropriate modification of SPSA for certain discrete optimization problems and state its basic properties. In particular we consider optimization problems where the value of the cost function can be evaluated only for *integer-valued variables*, is defined in terms of a probability or expectation and has no closed-form expression.

The motivation for the discrete algorithm stems from a class of discrete resource allocation problems. In broad terms, the problem is to distribute a finite number of resources, in discrete amounts, to finitely many users in such a way that the allocation optimizes some performance measure. Several settings in which this problem arises are: scheduling data and message transmissions in communication and computer networks, distributing a search effort to detect the location of a moving target, and determining the sizes of buffers in a manufacturing system. A common feature of these problems that makes them difficult to solve is the cardinality of the search space, which is large even in low dimensions.

We are going to develop a stochastic search algorithm on $\mathbb{Z}^p$, where $\mathbb{Z}$ is the set of integers. Before introducing the algorithm, we introduce some notation and review some results from [7] on the *fixed gain* SPSA method on $\mathbf{R}^p$, where both the size of the perturbation and the stepsize of the parameter update are fixed.

2 The problem formulation

Consider the following problem: given a function $L(.) = L(\theta)$, for $\theta \epsilon D$, where $D \subset \mathbf{R}^p$ is an open domain. However, this function is not explicitly known, but noise-corrupted measurements are available, given in the form

$$M(n, \theta, \omega) = L(\theta) + \varepsilon_n$$

where $\varepsilon_n = \varepsilon(n, \theta, \omega)$ is a random variable over some probability space $(\Omega, \mathcal{F}, \mathcal{P})$. The objective is to minimize L using only noise-corrupted measurements.

The function $L(.)$ is assumed to be three-times continuously differentiable within D and have a unique minimizing value in D, say θ^*. The measurement-noise process ε is a zero-mean, so-called L-mixing, uniformly (in θ) bounded process, which is smooth with respect to θ in an appropriate technical sense. L-mixing is an essential technical condition that apparently can not be relaxed. It can be defined as follows: first we say that an $\mathbf{R}^m$-valued stochastic processes (x_n) is *M-bounded* if for all $1 \leq q < \infty$

$$M_q(x) := \mathrm{E}^{1/q}|x_n(\theta)|^q < \infty.$$

If (x_n) is M-bounded we shall also write $x_n = O_M(1)$. Similarly if c_n is a positive sequence we write $x_n = O_M(c_n)$ if $x_n/c_n = O_M(1)$.

Let $(\mathcal{F}_n^+), n \geq 0$, be a monotone increasing family of σ-algebras, and $(\mathcal{F}_n^-), n \geq 0$, be a monotone decreasing family of σ-algebras. We assume that for all $n \geq 0$, $\mathcal{F}_n^+$ and $\mathcal{F}_n^-$ are independent. An $\mathbf{R}^m$-valued stochastic process (x_n), $n \geq 0$, is *L-mixing* with respect to $(\mathcal{F}_n^+, \mathcal{F}_n^-)$, if it is $\mathcal{F}_n^+$-adapted, M-bounded, and for any $1 \leq q < \infty$ we have

$$\Gamma_q(x) = \sum_{\tau=0}^{\infty} \gamma_q(\tau, x) < \infty,$$

where

$$\gamma_q(\tau, x) = \sup_{n \geq \tau} \mathrm{E}^{1/q}|x_n - \mathrm{E}(x_n|\mathcal{F}_{n-\tau}^+)|^q, \quad \tau > 0.$$

To estimate the gradient of L at θ we use simultaneous random perturbations. Letting k denote the iteration time, at time k we take a random vector

$$\Delta_k = (\Delta_{k1}, ..., \Delta_{kp})^T,$$

where Δ_{ki} is a (doubly-indexed) sequence of i.i.d. Bernoulli random variables, taking values $+1$ or -1 with equal probability 1/2. The Δ_{ki}'s are assumed to be independent of the noise process (ε_n). (Distributions other than Bernoulli are possible. See [12].)

In fixed gain SPSA the size of the perturbation is fixed, say to some $c > 0$. Let $D_0 \subset D$ be an appropriate compact, convex domain. For each $\theta \epsilon D_0$ we take two measurements

$$\begin{aligned} M_k^+(\theta) &= L(\theta + c\Delta_k) + \varepsilon(2k-1, \theta + c\Delta_k) \\ M_k^-(\theta) &= L(\theta - c\Delta_k) + \varepsilon(2k, \theta - c\Delta_k). \end{aligned}$$

Then the estimator of the gradient at time k and at θ is

$$H(k,\theta) = \left[\frac{M_k^+(\theta) - M_k^-(\theta)}{2c\Delta_{k1}}, \ldots, \frac{M_k^+(\theta) - M_k^-(\theta)}{2c\Delta_{kp}} \right]^T.$$

3 The fixed gain SPSA method

Before introducing the discrete SPSA algorithm, we review some results on fixed gain SPSA for continuous parameter optimization.

Let $a > 0$ be a fixed stepsize of the updating formula, called the gain. Starting with an initial estimate $\widehat{\theta}_1$, we compute recursively a sequence of estimated parameters, $\widehat{\theta}_k$, by

$$\widehat{\theta}_{k+1} = \widehat{\theta}_k - aH(k+1, \widehat{\theta}_k). \tag{1}$$

The assumed boundedness of the noise and the assumed stability of the so-called associated ODE ensures the boundedness of the sequence $\widehat{\theta}_k$. The pathwise behaviour of estimator processes generated by fixed gain SPSA methods can be analyzed using the result of [7]:

Theorem. *Under appropriate technical conditions, among others for good initial conditions*

$$|\widehat{\theta}_k - \theta^*| \leq \delta_k$$

where (δ_k) is an L-mixing process. In the small gain case with $a = \lambda, c = \lambda^{1/6}$ we have $\delta_k = O_M(\lambda^{1/3})$.

An improved estimator can be obtained using the averaged estimator sequence. Define

$$\overline{\widehat{\theta}}_k = \frac{1}{k}\sum_{i=1}^{k} \widehat{\theta}_i.$$

Corollary. *Under appropriate technical conditions and with $a = \lambda, c = \lambda^{1/6}$, λ small, we have with probability 1*

$$\limsup_{n \to \infty} |\overline{\widehat{\theta}}_k - \theta^*| = O(\lambda^{1/3}).$$

Another way of improving SPSA is to use ***higher order approximation*** of the gradient. For a function f having $2m+1$ continuous derivatives we have can approximate $f'(x)$ with an error of the order of

$$\frac{h^{2m+1}(-1)^m(m!)^2 f^{(2m+1)}(\xi)}{(2m+1)!}$$

(cf. [6]), which can be very small for even if we take $h = 1$ when f is sufficiently smooth. Higher order SPSA methods based on classical numerical differentiation schemes were developed and analyzed in [9]. Another possibility of improving efficiency is to use a *second-order* or Newton-type SPSA-method as proposed in [13, 14]. In the case of decreasing gains the asymptotic rate of convergence of Newton-type SPSA methods is slower than that of higher order SPSA methods, but for fixed gain procedures it may well be the other way round.

Assume now that θ is restricted to be integer-valued, i.e. $\theta \epsilon D \cap \mathbb{Z}^p$. Assume that L is convex in the sense that at any point of its graph there is a supporting hyperplane such that graph is on one side of this plane. Assume that there exists an extension of L to real-valued variables $\theta \epsilon D$, say $L^r(.) = L^r(\theta)$, so that the extended function is convex and sufficiently smooth. Then apply a suitably defined fixed gain SPSA method, with the additional caveat that we stay on the grid all the time. For this purpose we set

$$H^z(k,\theta) = [H(k,\theta)],$$

where $[x] = ([x_1],...,[x_p])$ and $[x_i]$ denotes the integer that less then or equal to x_i and closest to x_i, $1 \leq i < p$.

For the analysis of the resulting procedure we replace the function [.] by the smooth approximating function. Then the modified right hand side will be an L-mixing process, and [7] is applicable. Omitting the technical details, the viability of the procedure will be demonstrated by simulation results. The procedure can be extended to simple constrained optimization problems on grids.

4 Resource allocation

Our interest in SPSA on grids is motivated by multiple discrete resource allocation problems, which we briefly describe. The goal of discrete resource allocation is to distribute a finite amount of resources of different types to finitely many classes of users, where the amount of resources that can be allocated to any user class is discrete. Suppose there are n types of resources, and that the number of resources of type i is N_i. Resources of the same type are identical. The resources are allocated over M user classes: the number of resources of type i that are allocated to user class j is denoted by θ_{ij}. The matrix consisting of the θ_{ij}'s is denoted by Θ.

For each allocation Θ, there is an associated performance or reliability cost, which is denoted by $L(\Theta)$. We assume that the total cost is weakly separable in the following sense:

$$L(\Theta) = \sum_{j=1}^{M} L_j(\theta_j)$$

where $L_j(\theta_j)$ is the individual cost incurred by class j, $\theta_j = (\theta_{ij}, ..., \theta_{ij})$, i.e. the class j cost depends only on the resources that are allocated to class j. An important feature of resource allocation problems is that often the cost L_j is not given explicitly, but rather in the form of an expectation or in practical terms by simulation.

Then the discrete, multiple constrained resource allocation problem is:

$$\min L(\Theta)$$

subject to

$$\sum_{j=1}^{M} \theta_{ij} = N_i, \ \theta_{ij} \geq 0, \ 1 \leq i \leq n \qquad (2)$$

where the θ_{ij}'s are non-negative integers. We will assume that a solution exists with strictly positive components. Then the minimization problem is unconstrained on the linear manifold defined by the balance equations.

Problem (2) includes many problems of practical interest including the problem of optimally distributing a search effort to locate a moving target whose position is unknown and time varying (cf. [5]) and the problem of scheduling time slots for the transmission of messages over nodes in a radio network (cf. [3]). The above problem is a generalization of the single resource allocation problem with $m = 1$, considered in [2], where the total cost is separable.

Cassandras et al. [2] present a relaxation-type algorithm for the single resource, in which at any time the allocation is rebalanced between exactly two tasks. The continuous-variable version of their algorithm is as follows: for a pair of tasks (j,k) the new allocation vector θ^+ will differ in just two components from the previous value, which are given by

$$\begin{aligned} \theta_j^+ &= \theta_j + a(\frac{\partial}{\partial \theta_k} L_k(\theta_k) - \frac{\partial}{\partial \theta_j} L_j(\theta_j)) \\ \theta_k^+ &= \theta_k + a(\frac{\partial}{\partial \theta_j} L_j(\theta_j) - \frac{\partial}{\partial \theta_k} L_k(\theta_k)). \end{aligned}$$

Here a is a suitable stepsize. Obviously, the above rebalancing satisfies the balance equations. The selection of the pair (j,k) is done by a stochastic comparison method.

A stochastic version of the above algorithm is obtained if we replace $\frac{\partial}{\partial \theta_j} L_j(\theta_j)$ by their estimates obtained by simultaneous perturbation at time t, and denoted by $H_j(t,\theta_j)$. Thus we arrive to the following recursion: at time t select a pair (j,k) and then modify the allocation for this pair of tasks as follows:

$$\begin{aligned} \theta_{j,t+1} &= \theta_{j,t} + a(H_k(t,\theta_k) - H_j(t,\theta_j)) \\ \theta_{k,t+1} &= \theta_{k,t} + a(H_j(t,\theta_j) - H_k(t,\theta_k)), \end{aligned}$$

where a is a fixed gain. Obviously, the balance equations are not violated by the new allocation. The selection of the pair (j,k) can be done by a simple cyclic visiting schedule.

To ensure the non-negativity constraints we use a standard resetting mechanism. A new feature of the proposed algorithm is that it is *asynchronous* in the sense that only two components are updated at a time. Analysis of such procedures for very general, approximately Markovian visiting schedule for the pairs (j, k) has been given in [1] in the decreasing gain case (cf. condition (2.6) of the cited work). Taking $a = 1$ and replacing H by $[H]$ we get a stochastic approximation procedure searching over the grid of feasible allocations.

5 Simulation results

We present simulation results concerning fixed gain SPSA for randomly generated simple quadratic function $L(\theta)$ in $\mathbb{R}^{20}$ the minimal value of which is 0. In Figures 1–4 below we plot the value of the cost function vs. the iteration time for different (fixed) stepsizes $a = 0.01$ and $a = 1$ respectively, and the distance of the true minimum and the improved estimator obtained by averaging, i.e. $\bar{\bar{\theta}}_k$. In contrast to what is predicted by theory we had to add a resetting mechanism to ensure stability of the procedure. On Figure 5 and 6 the corresponding results are given, when the minimization over $\mathbb{Z}^{20}$ was considered.

6 Discussion

We have presented a fixed gain SPSA method and have given its basic theoretical properties. In contrast to by now standard weak-convergence results (cf. [10, 11]) our result is not of asymptotic nature. In fact it is applicable when the gain is fixed say to be equal to 1. Taking the size of the perturbation to 1 as well and truncating the estimated gradient we arrive to an SPSA-based estimator sequence that lives on a grid. An asynchronous version of this algorithm is very well suited for the solution of multiple resource allocation problems. The viability of the basic procedure is demonstrated by simulation examples.

Acknowledgement

This work was partially supported by the JHU/APL Independent Research and Development Program and by the National Research Foundation of Hungary (OTKA) under Grant T 020984.

References

[1] V. S. Borkar. Asynchronous stochastic approximations. *SIAM J. Control and Optimazation*, 36:840–851, 1998.

[2] C. G. Cassandras, L. Dai, and C. G. Panayiotou. Ordinal optimization for a class of deterministic and stochastic discrete resource allocation problems. *IEEE Trans. Automatic Control*, 43(7):881–900, 1998.

[3] C.G. Cassandras and V. Julka. Scheduling policies using marked/phantom slot algorithms. *Queueing Systems: Theory and Appl.*, 20:207–254, 1995.

[4] H.F. Chen, T.E. Duncan, and B. Pasik-Duncan. A stochastic approximation algorithm with random differences. In J.Gertler, J.B. Cruz, and M. Peshkin, editors, *Proceedings of the 13th Triennal IFAC World Congress, San Francisco, USA*, pages 493–496, 1996. Volume editors: R.Bitmead, J. Petersen, H.F. Chen and G. Picci.

[5] J.N. Eagle and J.R. Yee. An optimal branch-and-bound procedure for the constrained path, moving target search problem. *Operations Research*, 38, 1990.

[6] L. Fox. *Two-point boundary problems in ordinary differential eqautions.* Oxford at the Clarendon Press, 1957.

[7] L. Gerencsér. On fixed gain recursive estimation processes. *J. of Mathematical Systems, Estimation and Control*, 6:355–358, 1996. Retreival code for full electronic manuscript: 56854.

[8] L. Gerencsér. SPSA with state-dependent noise-a tool for direct adaptive control. In *Proceedings of the Conference on Decision and Control, CDC 37.* IEEE, 1998.

[9] L. Gerencsér. Rate of convergence of moments for a simultaneuous perturbation stochastic approximation method for function minimization. *IEEE Trans. Automatic Control*, 44:894–906, 1999.

[10] H.J. Kushner. *Approximation and Weak Convergence Methods for Random Processes.* MIT Press, 1984.

[11] H.J. Kushner and G. Yin. *Stochastic Approximation Algorithms and Applications.* Springer Verlag. New York, 1997.

[12] J.C. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Automatic Control*, 37:332–341, 1992.

[13] J.C. Spall. Accelerated second-order stochastic approximation algorithm using only function measurements. In *Proceedings of the 1997 IEEE CDC*, pages 1417–1424, 1997.

[14] J.C. Spall. Adaptive stochastic approximation by the simultaneous perturbation method. In *Proceedings of the 1998 IEEE CDC*, pages 3872 – 3879, 1998.

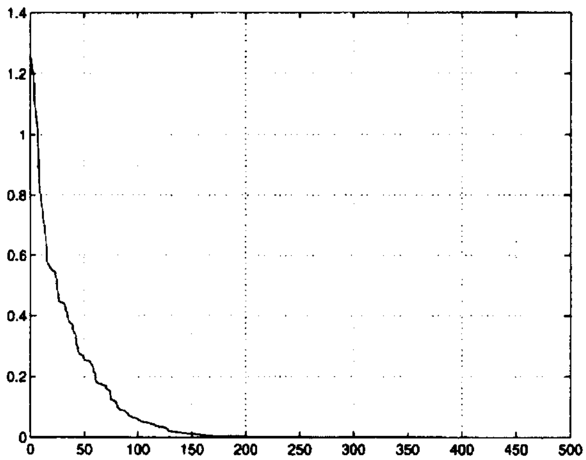

Figure 1: The value of $L(\widehat{\theta}_k)$, when $a = c = 0.01$

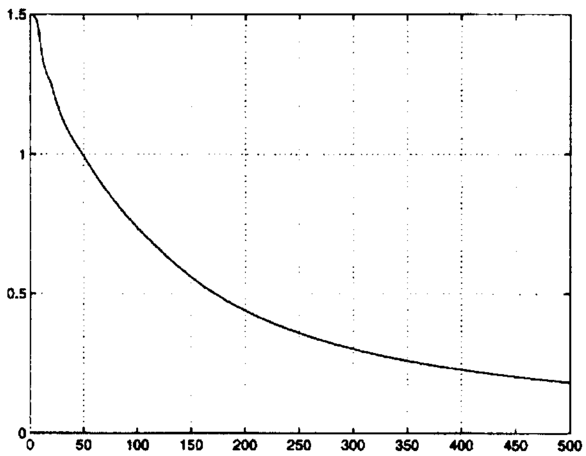

Figure 2: The distances $\|\bar{\widehat{\theta}}_k - \theta\|$, when $a = c = 0.01$

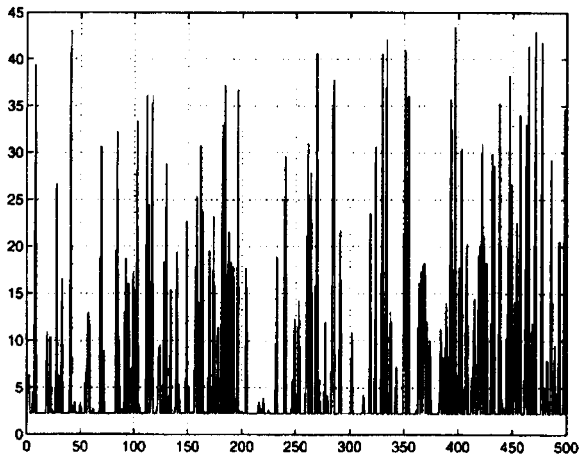

Figure 3: The value of $L(\widehat{\theta}_k)$, when $a = c = 1$

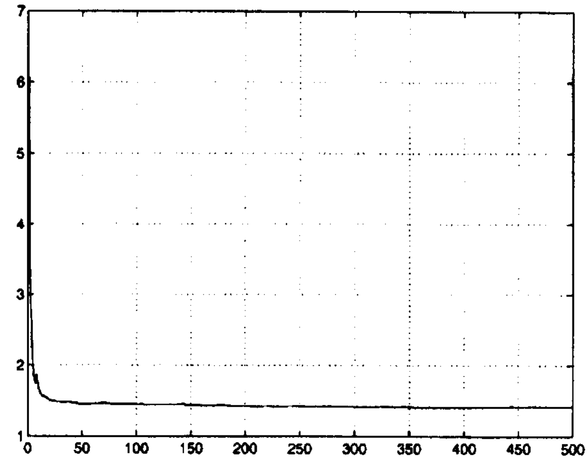

Figure 4: The distances $\|\bar{\widehat{\theta}}_k - \theta\|$, when $a = c = 1$

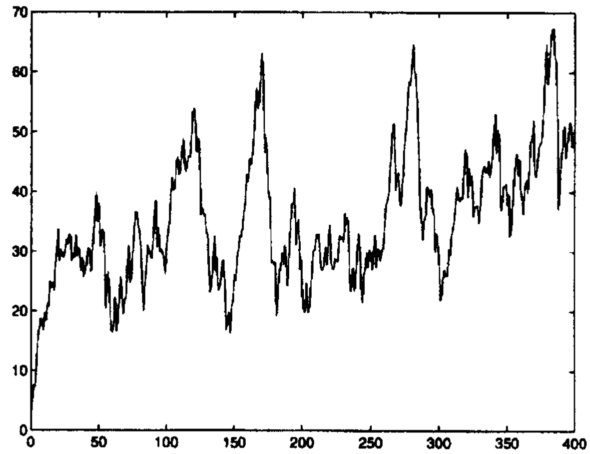

Figure 5: The value of $L(\widehat{\theta}_k)$, when $a = c = 1$, the minimization was made over $\mathbb{Z}^{20}$

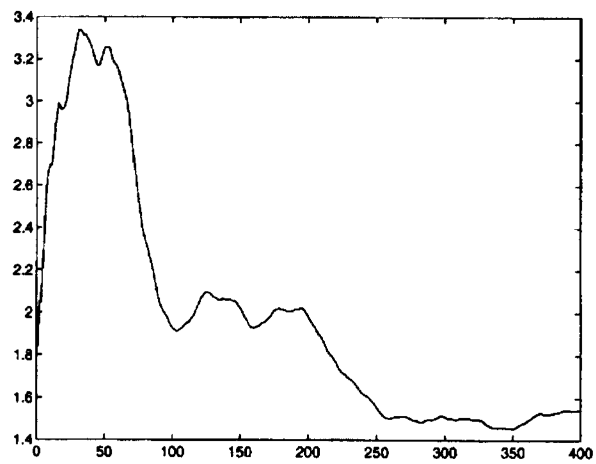

Figure 6: The distances $\|\bar{\widehat{\theta}}_k - \theta\|$, when $a = c = 1$, the minimization was made over $\mathbb{Z}^{20}$