ON SAMPLING-CONTROLLED STOCHASTIC APPROXIMATION

Paul Dupuis and Rahul Simha

COINS Technical Report 89-77

July 1989

On Sampling-controlled Stochastic Approximation[1]

PAUL DUPUIS
Department of Mathematics and Statistics
University of Massachusetts
Amherst, Mass. 01003

RAHUL SIMHA
Computer and Information Sciences Department
University of Massachusetts
Amherst, Mass. 01003

Abstract

In the general area of optimization under uncertainty, there are a large number of applications which require finding the 'best' values for a set of control variables or parameters and for which the only data available consist of measurements prone to random errors. Stochastic approximation, which provides a method of handling such noise or randomness in data, has been widely studied in the literature and is used in several applications. In this paper, we examine a new class of stochastic approximation procedures that are based on carefully controlling the number of observations or measurements taken before each computational iteration. This method, which we refer to as Sampling-controlled Stochastic Approximation, has advantages over standard stochastic approximation such as requiring less computation and the ability to handle bias in estimation. We explore the minimum growth rate required of the number of samples and prove a general convergence theorem for this new stochastic approximation method. In addition, we present applications to optimization and also derive a sampling-controlled version of the classic Robbins-Munro algorithm.

[1] This research has been supported by the Office of Naval Research under contract N00014-87-K-0304 and partially supported by the National Science Foundation under grant DMS-8902333.

1. Introduction

Stochastic approximation refers to a general technique for augmenting deterministic iterative algorithms in order to handle noise in the inputs. In the case of optimization, for example, there are several algorithms [29,35] that cause a collection of control variables to iteratively move towards an optimum value using values of some function (inputs) at each step. When these values cannot be analytically computed and, instead, must be *estimated*, stochastic approximation presents a viable method of using *measurements* (subject to random errors) in order to reach the optimum.

Ever since the introduction of the now classic Robbins-Munro algorithm [33], the stochastic approximation technique has found several applications [12,23,30] from bio-assay [45] to theories of learning [25] and has received wide attention in the literature [2,3,8,10,13,19,23,24,28,45]. An important theoretical concern in several research efforts is the convergence of the resulting algorithms to 'desired' values. This is accomplished, in the usual manner of stochastic approximation, by prescribing a sequence of *decreasing stepsizes* [23], one for each computational step of the algorithm. In this case, it can be shown that, for a wide class of algorithms, convergence to the desired value is obtained with probability one [3,8,23,28]. We refer to this method of augmenting an algorithm as *stepsize-controlled iteration.*

In this paper, we examine a new class of stochastic approximation procedures that achieve algorithmic convergence through repeated sampling of inputs while using a *fixed* stepsize. This procedure of adapting an algorithm in order to deal with noise in the inputs, has advantages over the standard method method of decreasing stepsizes such as requiring less computation [36] and handling inherent bias in measurements. In addition, it allows convergence results under somewhat different and, in several cases, weaker assumptions than those required for decreasing stepsize algorithms.

The ability of a stochastic approximation scheme to handle bias is particularly important in queueing systems and simulation methodology [10,12,13,27,30] since several estimators associated with queues are biased [15,21,41]. The stochastic approximation technique studied in this paper, referred to as *sampling-controlled iteration* was introduced by [36] in the context of load balancing for computer systems. The convergence result in [36] required several strong conditions and an unnecessarily high rate of sampling. A similar approach has been independently proposed in [44], although in [44], both repeated sampling and decreasing stepsizes are used and, in addition to stronger assumptions, a convergence result is shown using a nonstandard definition of convergence.

Similar ideas that weigh the number of samples of inputs against decreasing stepsizes have been introduced by other research efforts in stochastic approximation [5,19] as well as in other areas [1]. In [1], a fixed number of samples was used for an algorithm in the context of learning theory and experimental results were shown to demonstrate superiority over related algorithms using single samples. In [5], a fixed number of iterations was considered and the optimal selection of stepsizes was studied in order to minimize the mean-squared-error for the particular case of the Robbins-Munro algorithm [33]. These results, together with other examples of algorithmic speedup, are summarized in [19].

Our main contribution is a general convergence result. We explore the boundaries of the space of assumptions needed for convergence. One consequence is a convergence theorem with much weaker conditions than the one in [36]. In particular, the number of samples used at each iteration grows much more slowly. We demonstrate the utility of our method with applications to gradient-based optimization algorithms as well as an illustrative derivation of a sampling-controlled version of the classic Robbins-Munro algorithm. Our proofs, which are based on elementary upper bound results from Large Deviation Theory [43], are simpler than corresponding proofs for standard stochastic approximation.

In the next section, we consider a simple general recursion on a single variable in order to motivate and discuss our ideas. In Section 3 we present our convergence results for a general class of iterative algorithms on a multidimensional control variable. Finally, we present some applications in Section 4 before making our concluding remarks in Section 5.

2. Sampling-controlled Stochastic Approximation

For motivational purposes, we first examine the following simplified iteration on a single control variable:

$$x^{(n+1)} = x^{(n)} + a_n h(x^{(n)}) \tag{1}$$

where n denotes the iteration number, h is a real-valued function and a_n is a real-valued stepsize for each n. We refer to the above iteration as the *deterministic version* of the algorithm h. This framework, we note, is representative of several root-finding iterative algorithms over a single variable (such as Newton's Method [35]) and, in these cases, h is computed analytically. Next, let x^* be some desired point of convergence such that $\lim_{n\to\infty} x^{(n)} = x^*$ with $x^{(n)}$ defined above in equation (1).

Let $\left\{\hat{Y}^i(x), x \in R\right\}$ be a family of random variables and define

$$\hat{Y}^{L(n)}(x) = \frac{1}{L(n)} \sum_{i=1}^{L(n)} \hat{Y}^i(x)$$

to be a point estimate [34] of $Y(x)$ based on $L(n)$ independent, identically distributed random samples, $\hat{Y}^1(x), ..., \hat{Y}^{L(n)}(x)$. Typically, $Y(x)$ represents a system quantity that is relatively easy to estimate whereas $h(x)$ is a more complex system property that is expressed as a function of simpler quantities such as $Y(x)$. Then, in order to estimate h, it becomes necessary to use an estimate, $g\left(\hat{Y}^{L(n)}(x)\right)$, constructed from an estimate of $Y(x)$, where g is a function on the range of $\hat{Y}^i(x)$. We will first consider the simple case in which g is the identity function (i.e., $E\left[g\left(\tilde{Y}^L(x)\right)\right] = h(x)$). and later, allow g to be a more complicated, nonlinear function.

Let the stepsizes satisfy the standard conditions [23,33]: $a_n > 0, a_n \to 0$ and $\sum_{i=1}^{\infty} a_i = \infty$. Then, it is possible to show that [3,8,23,28], under very general conditions, when $\forall n : L(n) = L > 0$, the recursion defined by:

$$x^{(n+1)} = x^{(n)} + a_n g\left(\tilde{Y}^L(x^{(n)})\right) \tag{2}$$

converges to x^*, i.e., $\lim_{n\to\infty} x^{(n)} = x^*$ *a.s.* (the notation *a.s.* is used to denote *almost surely* or convergence with probability one [34]). We refer to an iteration (related to h) involving random variables, such as $\tilde{Y}^L(x^{(n)})$, as the *stochastic version* of algorithm h and to the particular form of the stochastic version in (2) as the *stepsize-controlled* version of h. This is the approach taken in standard stochastic approximation [23] and we distinguish this from our method of stochastic approximation.

We focus on a particular subclass of deterministic recursions of the type in equation (1) with $\forall n : a_n = a$ and where $\exists a > 0$ such that $\lim_{n\to\infty} x^{(n)} = x^*$ with $x^{(n)}$ defined by equation (1). Thus, we are concerned with the subclass of *deterministic* recursions,

$$x^{(n+1)} = x^{(n)} + ah(x^{(n)}), \tag{3}$$

in which algorithmic convergence to the desired value is obtained with a *fixed* stepsize. We note that there is large class of useful numerical and optimization algorithms [29,35] in several recent applications [4,11,39,46] which satisfy the above property. A *stochastic* version is easily obtained by augmenting (3) with decreasing stepsizes that satisfy the usual conditions on stepsizes [23]: $a_n > 0, a_n \to 0$ and $\sum_{i=1}^{\infty} a_i = \infty$:

$$x^{(n+1)} = x^{(n)} + a_n g\left(\tilde{Y}^L(x^{(n)})\right) \tag{4}$$

Once again, we obtain a stepsize-controlled *stochastic* algorithm for which it may by shown that $\lim_{n\to\infty} x^{(n)} = x^*$ *a.s.* [8,23]. We note that equation (4) is obtained from equation (3) with the fixed stepsize, a, replaced by decreasing stepsizes, a_n, and it is for this reason that we refer to standard stochastic approximation as stepsize-controlled iteration. Although equations (2) and equations(4) are identical, we repeat (4) in order to emphasize the difference between the derivations of the two equations. In particular, we observe that since convergence is obtained with a fixed stepsize in the deterministic case, the latter recursion uses decreasing stepsizes only for the purpose of removing noise. This observation is central to our method of stochastic approximation: we replace noise removal through decreasing stepsizes with noise removal via increasing sampling.

In several applications, for example the load balancing problem in [36], the estimator $g\left(\tilde{Y}^L(x)\right)$ may be biased, i.e., for any fixed L, $E\left[g\left(\tilde{Y}^L(x)\right)\right] \neq h(x)$ [34]. It is known, also, that most estimators associated with queueing systems suffer from this small-sample bias [15,21,41]. This bias is usually due to the nonlinearity of the function g and the correlation present in a regenerative cycle of a queue. In these cases, although $x^{(n)}$ defined by equation (4) converges, $\lim_{n\to\infty} x^{(n)} = x' \neq x^*$. More seriously, in the case of optimization problems such as load balancing [36], the limit x' may not even be a *feasible* solution [36], i.e., may not satisfy given constraints on x'. Most often, $g\left(\tilde{Y}^L(x^{(n)})\right)$ represents a measurement taken from a system after $x^{(n-1)}$ has been changed to $x^{(n)}$. In several situations [42], h is a desired steady-state expectation and when L is too small, the system is not given enough 'settling' time [42], thereby causing the estimator $g\left(\tilde{Y}^L(x^{(n)})\right)$ to record effects of transient behavior. In this case, which is also representative of many queueing systems, we have $E\left[g\left(\tilde{Y}^L(x^{(n)})\right)\right] \neq h(x^{(n)})$.

With this motivation, we consider an alternate form of stochastic approximation, in which the stepsize, a, remains fixed and, instead, the number of samples, $L(n)$, increases to infinity. We define the following *stochastic* iteration:

$$x^{(n+1)} = x^{(n)} + ag\left(\tilde{Y}^{L(n)}(x^{(n)})\right) \tag{5}$$

where $L(n)$ is a sequence of positive integers such that $L(n) \to \infty$. Intuitively, as $L(n) \to \infty$, we would expect that for large n, the procedure tends to behave like its deterministic counterpart, equation (3). Now, if $g\left(\tilde{Y}^{L(n)}(x)\right)$ is a *strongly consistent* estimator [34] of $h(x)$, i.e.,

$$\lim_{n\to\infty} g\left(\tilde{Y}^{L(n)}(x)\right) = h(x) \quad a.s.$$

(but not necessarily unbiased), then, as n gets larger, the effects of bias are slowly removed. In this manner, we might obtain the desired convergence, $\lim_{n\to\infty} x^{(n)} = x^*$ $a.s.$.

In the next section, we prove a general convergence theorem for a multidimensional recursion on $X^{(n)} = (x_1^{(n)}, ..., x_K^{(n)})$. We derive sufficient conditions on the growth of $L(n)$ based on assumptions made about the estimators and the algorithm that guarantee convergence to a desired point, X^*, almost surely.

3. Convergence

In order to prove our convergence result we use the following notation:

- For $i \in \{1, ..., p\}$ and fixed $X \in D \subset R^K$, define

$$\tilde{Y}_i^m(X) = \frac{1}{m}\sum_{j=1}^{m} \hat{Y}_i^j(X)$$

where $\hat{Y}_i^1(X), \hat{Y}_i^2(X), ..., \hat{Y}_i^j(X), ...$ are independent and identically distributed random variables with mean $Y_i(X)$. Then, under weak conditions on the distributions of $\hat{Y}_i^j(X)$, the Strong Law of Large Numbers [34] implies

$$\lim_{m\to\infty} \tilde{Y}_i^m(X) = Y_i(X) \;\; a.s.$$

- For $i \in \{1, ..., K\}$ let $g_i(y_1, ..., y_p)$ be continuous functions on R^p and define

  $$h_i(X) = g_i\left(Y_1(X), ..., Y_p(X)\right)$$

  Then, since each g_i is continuous,

  $$\lim_{m\to\infty} g_i\left(\tilde{Y}_1^m(X), ..., \tilde{Y}_p^m(X)\right) = h_i(X) \;\; a.s.$$

- Finally, consider the following *fixed-stepsize* deterministic algorithm, defining $X^{(n+1)} = (x_1^{(n+1)}, ..., x_K^{(n+1)})$ in terms of $X^{(n)}$. For each $i \in \{1, ..., K\}$:

  $$x_i^{(n+1)} = x_i^{(n)} + aH_i\left(h_1(X^{(n)}), ..., h_K(X^{(n)}), X^{(n)}\right) \tag{6}$$

  where each $H_i, i \in \{1, ..., K\}$ may or may not be continuous. We will henceforth also use g and h to denote the vectors $(g_1, ..., g_K)$ and $(h_1, ..., h_K)$.

We focus on deterministic recursions such that $\exists a > 0$ for which it is true that $\forall X^{(0)}$:

$$\lim_{n\to\infty} X^{(n)} = X^* \tag{7}$$

where X^* is the desired point of convergence. We will show that the *sampling-controlled stochastic* version of equation (6),

$$x_i^{(n+1)} = x_i^{(n)} + aH_i\left(g\left(\tilde{Y}_1^{L(n)}(X^{(n)}), ..., \tilde{Y}_p^{L(n)}(X^{(n)})\right), X^{(n)}\right) \tag{8}$$

converges to X^*. In comparing equations (6) and (8), we observe that the functions $h_i(X^{(n)})$ in equation (6) have been replaced by the corresponding estimators $g_i\left(\tilde{Y}^{L(n)}(X^{(n)})\right)$ based on $L(n)$ samples at iteration n. We will assume throughout the paper that the procedure remains in a compact set $D \subset R^K$. This may have to be accomplished through some type of constraint mechanism [11,35,37], which we will assume to be incorporated into the functions H_i.

Assumptions:

*A*1. *Existence of moment generating functions.* We assume that

$$\sup_{X \in D} E\left[\exp\left(\sum_{i=1}^{p} \alpha_i \left(\hat{Y}_i^j(X) - Y_i(X)\right)\right)\right]$$

is finite in some open neighbourhood of $\alpha = (\alpha_1, ..., \alpha_p) = 0$ with a derivative at $\alpha = 0$.

*A*2. *Continuity of g.* We assume that the functions $g_i(y_1, ..., y_p)$, $i \in \{1, ..., K\}$, are continuous in the domain of interest, i.e., in the range of $\tilde{Y}_1^m(X), ..., \tilde{Y}_p^m(X)$.

*A*3. *Stability of convergence under perturbations.* We assume that the deterministic version given by equation (6) satisfies the following stability property. Given $\delta > 0$ there exists $N < \infty$ and $\epsilon > 0$ such that if

$$x_i^{(n+1)} = x_i^{(n)} + aH_i\left(h_1(X^{(n)}) + \epsilon_1^n, ..., h_K(X^{(n)}) + \epsilon_K^n, X^{(n)}\right) \tag{9}$$

and if $\sup_{0 \leq n \leq N} |\epsilon_i^n| < \epsilon$ for all $i \in \{1, ..., K\}$ then for all $X^{(0)} \in D$ and all $N \leq n \leq 2N$,

$$\left|X^{(n)} - X^*\right| \leq \delta$$

Intuitively, this last assumption characterizes the convergence behavior of the deterministic version under infinitesimally small perturbations by requiring the trajectory of the perturbed version, equation (9), to behave nearly as well as the unperturbed version, equation (6), for a certain fixed number of iterations (N to $2N$), for small enough perturbations (smaller than ϵ).

THEOREM 1: There exists a lower semi-continuous convex function $L_Y(\beta) : R^p \to [0, \infty]$ with the properties

(i) $L_Y(\beta) = 0$ if and only if $\beta = (Y_1(X), ..., Y_p(X))$.
(ii) The sets $\{\beta : L_Y(\beta) \leq s\}$ are compact for all $s \in [0, \infty)$.
(iii) Given $\delta > 0$ and $s > 0$ there exists $M < \infty$ such that uniformly in $X \in D$ and A satisfying $\inf_{\beta \in A} L_Y(\beta) \geq s$,

$$P\left[\left(\tilde{Y}_1^m(X), ..., \tilde{Y}_p^m(X)\right) \in A\right] \leq \exp\left(-(s - \delta)m\right)$$

for all $m \geq M$.

PROOF: (see Appendix).

THEOREM 2: There exists a lower semi-continuous convex function $L_g(\gamma) : R^K \to [0, \infty]$ with the properties

(i) $L_g(\gamma) = 0$ if and only if $\gamma = h(X)$.

(ii) The sets $\{\gamma : L_g(\gamma) \leq s\}$ are compact for all $s \in [0, \infty)$.

(iii) Given $\delta > 0$ and $s > 0$ there exists $M < \infty$ such that uniformly in $X \in D$ and A satisfying $\inf_{\gamma \in A} L_g(\gamma) \geq s$,

$$P\left[g\left(\tilde{Y}_1^m(X), ..., \tilde{Y}_p^m(X)\right) \in A\right] \leq \exp\left(-(s-\delta)m\right)$$

for all $m \geq M$.

(iv) The function $L_g(\gamma)$ can be expressed in terms of $L_Y(\beta)$:

$$\begin{aligned} L_g(\gamma) &= \inf\{L_Y(\beta) : g(\beta) = \gamma\}, \quad \text{if } g(\beta) = \gamma \text{ has a solution} \\ & \quad +\infty, \quad \text{otherwise} \end{aligned}$$

PROOF: (See Appendix).

THEOREM 3: Under assumptions $A1 - A3$, the stochastic version, equation (8), converges to X^*, i.e.,

$$\lim_{n \to \infty} X^{(n)} = X^* \quad a.s.$$

if $L(n) = c_n \log n$ where $c_n \to \infty$ as $n \to \infty$.

PROOF: (See Appendix).

3.1 Discussion

The proof of our convergence result may be interpreted by some simple intuitive arguments. First, we recall a few facts from large deviation theory. For fixed X, the estimator $\tilde{Y}_i^m(X)$ converges to the mean $Y_i(X)$ via the Strong Law of Large Numbers. In this case elementary Large Deviation Theory (Cramer's Theorem in [43]) provides a simple upper bound on the probability that $\tilde{Y}_i^m(X)$ is found in any closed set A not containing $Y_i(X)$ after m samples:

$$P\left[\tilde{Y}_i^m \in A\right] \leq \exp\left(-c(A)m\right) \tag{10}$$

where $c(A) > 0$ is a number derived from A (the large deviation rate [43]). Conversely, the Borel-Cantelli lemma [34] may be used to show that if an estimator satisfies equation (10), i.e., possesses a large deviation property, then it is converges. Also, large deviation properties extend to continuous functions of $\tilde{Y}_i^m(X)$, as in Theorem 2.

Next we observe that the recursion given by equation (8) may be viewed as an *estimator* for X^* and hence if a large deviation property could be derived for $X^{(n)}$, the Borel-Cantelli lemma could be used in proving convergence. This would in fact be quite simple to achieve if the functions H_i were also continuous. However, many of the deterministic algorithms considered in the literature do not have this property. Hence we take a slightly different approach, which requires our use of the stability of convergence assumption. We have explicitly separated our study of the behavior of the estimator, g, from that of the algorithm, H, which may be discontinuous.

In deriving our results we have made use of assumption $A1$ for Theorem 1, $A2$ for Theorem 2 and $A3$ for Theorem 3 and we now discuss some of the implications of our assumptions. Assumption $A1$ is a statement on the behavior of the moments of $\hat{Y}_i^m$ and is stronger than assuming simply the existence of the moments. However, we note that stronger assumptions have often been made in the case of stepsize-controlled iteration [8,33]. In the case of gradient estimators for queues, such as the one described in section 4.1 of this paper, we believe that if the queue is recurrent and if the moments of the length and number in a busy period satisfy the assumption, then the moments of the gradient estimator, being polynomially-bounded functions of the busy period variables, will also satisfy the assumption.

Assumption $A2$ is a simple continuity assumption on the construction of more complicated estimators, g_i, from the more fundamental estimators $\tilde{Y}_i^m$. This assumption is easily verified for the gradient estimators of interest here [16,32], as well those used in other recursions [33], and appears not to be a restrictive assumption. Finally, assumption $A3$, which is a condition on the convergence behavior of the deterministic version, equation (6), reduces to a characterization of the functions H_i. Consider what would happen if this condition were not satisfied for some deterministic algorithm. Let $X^{(1)}, ..., X^{(n)}$ be a sequence generated by the algorithm starting at $X^{(0)}$ and let $Y^{(1)}, ..., Y^{(n)}$ be the sequence generated when infinitesimally small perturbations are added to each $Y^{(i)}$. In this case, violation of assumption $A3$ implies that arbitrarily small perturbations can cause $X^{(n)}$ and $Y^{(n)}$ to be radically different in the sense that, while $X^{(n)}$ generates 'good' values of the objective function, $Y^{(n)}$ results in poor performance. We argue that such algorithms are likely to be rejected in practice since their convergence behavior can be very different for arbitrarily small measurement errors. In the case of algorithms used in practice, we note that if the functions H_i are continuous then $A3$ is automatically satisfied and this includes several numerical procedures [29,35]. Furthermore, when H_i is not continuous [11], proofs of deterministic convergence usually consist of showing a strict improvement of the objective function and, in these cases, the individual proof would have to be modified to show that assumption $A3$ is satisfied. We also mention that, when multiple optima are present, assumption $A3$ should be rephrased to imply that the perturbed version satisfies the property of producing arbitrarily close values of the objective function.

Finally, some comments about the growth rate $L(n)$. Conditions on $L(n)$ arise naturally from our method of proof and the growth rate needed here is far slower than $L(n) > L(n-1)$, the rate required in [36]. Furthermore, in [44], decreasing stepsizes were used in addition to increased sampling and convergence was shown according to a nonstandard definition of convergence. We have demonstrated that a fixed stepsize is sufficient for strong convergence in the usual sense. We believe that $L(n) = c_n \log n$ is the slowest growth rate possible since with a slower growth rate such as $L(n) = \log n$ the Borel-Cantelli lemma cannot to be applied.

4. Applications

In this section we present two applications of our sampling-controlled methods to established algorithms. The first application is to a well-known routing algorithm [11] and the second, solely for illustrative purposes, is a sampling-controlled version of the classic Robbins-Munro algorithm [33].

4.1 A Gradient-based Routing Algorithm

We examine the application of our sampling-control technique to a well-known gradient-based hill-climbing algorithm, Gallager's algorithm [11], that has found applications in routing in computer networks [4,6,7,11], load balancing [26,31,37] as well as in the area of learning automata [38]. We focus on augmenting this algorithm into a stochastic approximation procedure using sampling-control. We note that the algorithm is gradient-based and thus, the *deterministic* version uses analytic formulas for the gradient at each iterative step. Furthermore, we observe that several methods for direct gradient estimation in queueing systems have recently received a great deal of attention in the literature [16,17,32] and therefore, a *stochastic* version of Gallager's algorithm, using these gradient estimates, is of general interest in the above application areas.

We concern ourselves with applications in which direct gradient estimation is possible and has already been used. These examples include the optimization of parameters in a queueing network [18], load balancing [31,36] and routing [6,7], some of which, we note, have considered the use of stepsize-control [18,22,23]. In order to demonstrate the use of sampling-control, we examine a particular application - the optimization of arrival rates in a queueing network - and develop a sampling-controlled extension of Gallager's algorithm that uses estimates of the derivatives of queueing delays (with respect to these arrival rates) in order to minimize the overall mean delay. We will employ the *likelihood-ratio* estimation method in [32] and describe how derivative estimates may be obtained and used in Gallager's gradient-based optimization algorithm.

Let us define the following notation associated with a single queue in a queueing network [32]:

N_j = the number of customers in the j^{th} busy period [20].

W_{ij} = the waiting time of the i^{th} customer in the j^{th} busy period.

$W_j = \sum_{i=1}^{N_j} W_{ij}$

λ = the arrival rate to the queue. It is assumed that the arrival process is Poisson [20].

T_j = the duration of the j^{th} busy period.

D = the expected steady-state waiting time for the queue.

For a fixed arrival rate, λ, we describe the estimation scheme of [32] and formally define the following likelihood-ratio estimators:

$$\begin{aligned}
\tilde{Y}_1^{L(n)}(\lambda) &= \frac{1}{L(n)} \sum_{j=1}^{L(n)} W_j \\
\tilde{Y}_2^{L(n)}(\lambda) &= \frac{1}{L(n)} \sum_{j=1}^{L(n)} \left(\frac{N_j}{\lambda} - T_j\right) W_j \\
\tilde{Y}_3^{L(n)}(\lambda) &= \frac{1}{L(n)} \sum_{j=1}^{L(n)} \left(\frac{N_j}{\lambda} - T_j\right) N_j \\
\tilde{Y}_4^{L(n)}(\lambda) &= \frac{1}{L(n)} \sum_{j=1}^{L(n)} N_j
\end{aligned}$$

Note that each one of the the series of $L(n)$ samples in the above estimators is obtained from a busy period of the queue and hence are independent [21]. Then, from this regenerative property and the strong law [32], we have, for $1 \leq i \leq 4$,

$$\lim_{n \to \infty} \tilde{Y}_i^{L(n)}(\lambda) = E\left[\tilde{Y}_i^1(\lambda)\right]$$

Next, let

$$\hat{g}\left(\tilde{Y}_1^{L(n)}(\lambda), ..., \tilde{Y}_4^{L(n)}(\lambda)\right) = \frac{\tilde{Y}_2^{L(n)}(\lambda)}{\tilde{Y}_4^{L(n)}(\lambda)} - \frac{\tilde{Y}_3^{L(n)}(\lambda)\tilde{Y}_1^{L(n)}(\lambda)}{\left(\tilde{Y}_4^{L(n)}(\lambda)\right)^2} \tag{11}$$

In [32], it is shown that $\hat{g}$, as defined above, is a strongly consistent estimator of the derivative of the mean delay of the queue with respect to the arrival rate, i.e.,

$$\lim_{n \to \infty} \hat{g}\left(\tilde{Y}_1^{L(n)}(\lambda), ..., \tilde{Y}_4^{L(n)}(\lambda)\right) = \frac{dD}{d\lambda} \ a.s.$$

However, we note that this estimator is biased due to the nonlinearity of $\hat{g}$ [15]. Now, in queueing network composed of K queues, an overall delay function can be defined as a continuous function, $C(D_1, ..., D_K)$ of the delays in each of the K individual queues and thus, derivative estimators, g_i, for each derivative $\frac{\partial C}{\partial \lambda_i}$, $1 \leq i \leq K$, may be expressed in terms of functions $\hat{g}_i$ such that for a fixed arrival rate λ_i at each queue:

$$\lim_{n \to \infty} g_i\left(\tilde{Y}_{i1}^{L(n)}(\lambda_i), ..., \tilde{Y}_{i4}^{L(n)}(\lambda_i)\right) = \frac{\partial C}{\partial \lambda_i}$$

We now define the gradient algorithm in terms of $\Lambda^{(n)} = (\lambda_1^{(n)}, ..., \lambda_K^{(n)})$ where $\lambda_i^{(n)}$ is the i^{th} parameter or i^{th} component of the multidimensional control variable, $\Lambda^{(n)}$, and where, n denotes the iteration number. In keeping with the notation defined in the previous section we let $h_i(\lambda)$ denote $\frac{\partial C}{\partial \lambda_i}$. Then, the *deterministic* version of the algorithm will be completely specified by listing expressions for $H_i\left(h_1(\lambda_1^{(n)}), ..., h_K(\lambda_K^{(n)})\right)$ in the recursion given in equation (6) with $X^{(n)}$ replaced by $\Lambda^{(n)}$. Let $m = \arg\min_{1 \leq i \leq K} h_i(\lambda_i^{(n)})$ i.e., $h_m(\lambda_m^{(n)}) = \min_{1 \leq i \leq K} h_i(\lambda_i^{(n)})$. Then define the following functions, H_i, which characterize the deterministic algorithm:

$$H_i\left(h_1(\lambda_1^{(n)}), ..., h_K(\lambda_K^{(n)})\right) = h_m(\lambda_m^{(n)}) - h_i(\lambda_i^{(n)}) \quad \forall i \neq m$$

$$H_m\left(h_1(\lambda_1^{(n)}), ..., h_K(\lambda_K^{(n)})\right) = \sum_{1 \leq i \leq K, i \neq m} h_i(\lambda_i^{(n)}) - h_m(\lambda_m^{(n)}) \qquad (12)$$

It can be shown that [4,11], under certain smoothness conditions on C, if Λ^* is the unique optimum solution that minimizes the delay function C, then

$$\lim_{n \to \infty} \Lambda^{(n)} = \Lambda^*$$

when $X^{(n)}$ is defined by the recursion, $\forall i : x_i^{(n+1)} = x_i^{(n)} + aH_i$.

For the applications in which such gradient-based optimization algorithms are employed, there are usually contraints on the control variables, $X^{(n)}$. In the routing [11] and load balancing [37] problems, for example, the control variables represent probabilities and hence $X^{(n)}$ is restricted to the simplex defined by: $\sum_{i=1}^{K} x_i^{(n)} = 1$ and $\forall i : 0 \leq x_i^{(n)} \leq 1$. The algorithm defined in equation (12) satisfies the first (equality) constraint and is easily modified [11] in order to handle the latter (inequality) constraint. Below, we directly present the *stochastic* version of the algorithm modified to handle both these types of constraints. In order to simplify the notation needed we define

$$\tilde{g}_i^{(n)} = g_i\left(\tilde{Y}_{i1}^{L(n)}(\lambda_i^{(n)}), ..., \tilde{Y}_{i4}^{L(n)}(\lambda_i^{(n)})\right)$$

and obtain the *stochastic* version as:

$$\lambda_i^{(n+1)} = \lambda_i^{(n)} + a\left(\tilde{g}_m^{(n)} - \tilde{g}_i^{(n)}\right) \qquad \forall i \in A^{(n)}$$

$$\lambda_m^{(n+1)} = \lambda_m^{(n)} + a \left(\sum_{i \in A^{(n)} i \neq m} \tilde{g}_i^{(n)} - \tilde{g}_m^{(n)} \right) \tag{13}$$

where $A^{(n)} = \{i | \lambda_i^{(n)} + a \left(g_m^{(n)} - g_i^{(n)} \right) > 0\} \cup \{m\}$ and $m = \arg\min_i \tilde{g}^{(n)}$

From equation (11), we observe that assumption $A2$ of section 3 is satisfied, i.e., g is a continuous function. Therefore, if we assume the existence of the moment generating function for the estimators $\hat{Y}_i^m(\lambda)$ then Theorems 1 and 2 are easily proven. Assuming that $A3$ is also satisfied, we may easily apply Theorem 3 to conclude that

$$\lim_{n \to \infty} \Lambda^{(n)} = \Lambda^*$$

Since the functions H_i are discontinuous, the verification of $A3$ for this algorithm depends on the application and, in particular, on the function C.

4.2 The Robbins-Munro Algorithm

In this section, for illustrative purposes, we examine a sampling-controlled stochastic analog of the Robbins-Munro algorithm [33] for cases in which the deterministic version converges with a fixed stepsize. In particular, we consider the following *deterministic* recursion on a single variable in a compact set:

$$x^{(n+1)} = x^{(n)} - a \left(f(x^{(n)}) - \theta \right) \tag{14}$$

where $f(x^*) = \theta$ and x^* is the desired point of convegence, i.e., $\lim_{n \to \infty} x^{(n)} = x^*$. In this case if we define the operator $Tx = x - a(f(x) - \theta)$, then in several instances [35] T can be shown to be a contraction mapping [35] with a unique fixed point at x^* and, therefore, $\lim_{n \to \infty} x^{(n)} = x^*$ in equation (14).

Next, let $\bar{y}^m(x^{(n)})$ be an estimator based on m random samples, $\hat{Y}^1(x^{(n)}), .., \hat{Y}^m(x^{(n)})$, such that

$$\bar{y}^m(x^{(n)}) = \frac{1}{m} \sum_{j=1}^{m} \hat{Y}^j(x^{(n)})$$

and

$$\lim_{m \to \infty} \bar{y}^m(x^{(n)}) = f(x^{(n)})$$

Further, by letting $\tilde{g}^{(n)} = \bar{y}^{L(n)}(x^{(n)})$ we have the following *sampling-controlled stochastic* version:

$$x^{(n+1)} = x^{(n)} - a \left(\tilde{g}^{(n)} - \theta \right) \tag{15}$$

Then, since $A1 - A3$ are satisfied, we have, by Theorem 3, $\lim_{n \to \infty} x^{(n)} = x^*$.

5. Conclusions

In this paper, we have presented a new stochastic approximation scheme, *sampling-controlled iteration*, that has the important capability of handling bias in estimators. Estimator bias is found in several problem areas and, consequently, sampling-controlled stochastic approximation finds application in related optimization, numerical and control algorithms. Since other research efforts [36] have also experimentally determined the usefulness of this method, we conclude that sampling-control offers an attractive alternative to the standard stochastic approximation method.

Our main contribution was a general convergence theorem which explored the bounds on growth rates of sampling required for convergence. Our proof methods, which are simple and accessible, use elementary large deviation bounds and easily determine the conditions on the rate of sampling. In addition to the convergence result, we apply our methods to an optimization problem in a queueing system. Furthermore, this being a relatively new procedure for stochastic approximation, we illustrate our method by presenting sampling-controlled version of the classic Robbins-Munro algorithm. For future work, it would be interesting to compare the adaptivity of modified stepsize- and sampling-controlled algorithms. Finally, we observe that although our focus has been on theoretical issues of convergence, much work needs to be done in terms of the application and empirical study of stochastic approximation methods.

REFERENCES

[1] A.G.Barto and M.I.Jordan, "Gradient Following Without Back-propogation in Layered Networks", *Proc. IEEE First Annual Conf. on Neural Networks*, San Diego, California, 1987.

[2] A.Becker, P.R.Kumar and C-Z.Wei, "Adaptive Control with the Stochastic Approximation Algorithm: Geometry and Convergence", *IEEE Trans. on Automatic Control*, Vol. AC-30, No. 4, April 1985, pp. 330-338.

[3] A.Benveniste, M.Metivier and P.Priouret, Algorithmes Adaptifs et Approximations Stochastiques, *Masson*, 1987.

[4] D.Bertsekas, E.Gafni and R.Gallager, "Second Derivative Algorithms for Minimum Delay Distributed Routing in Networks," *IEEE Trans. on Communications*, Vol COM-32, No. 12, pp. 911-919, August 1984.

[5] H.D.Block, "Estimates of Error for Two Modifications of the Robbins-Munro Stochastic Approximation Process", *Ann. Math. Stat.*, Vol. 28, pp. 1003-1010.

[6] C.Cassandras, V.Abidi and D.Towsley, "Distributed Routing with On-line Marginal Delay Estimation", *Proc. 8th Int. Conf. on Distributed Computing Systems*, San Jose, 1988.

[7] F.Chang and L.Wu, "An Optimal Adaptive Routing Algorithm", *IEEE Trans. on Automatic Control*, pp. 690-700, Vol. AC-31, No. 8, Aug 1986.

[8] P.Dupuis and H.J.Kushner, "Stochastic Approximation and Large Deviations: Upper Bounds and w.p.1 Convergence", *SIAM J. Control and Opt.*, to appear.

[9] R.C.Ellis, "Large Deviations for a General Class of Random Vectors", *Annals of Probability*, Vol. 12, No. 1, pp. 1-12, 1984.

[10] M.C.Fu and Y.C.Ho, "Using Perturbation Analysis for Gradient Estimation, Averaging and Updating in a Stochastic Approximation Algorithm", *Proc. Winter Simulation Conf.*, 1988, pp. 509-517.

[11] R.G.Gallager, "A Minimum Delay Routing Algorithm Using Distributed Computation," pp. 73-85, *IEEE Trans. on Communications*, Vol COM-25, 1977.

[12] P.W.Glynn, "Optimization of Stochastic Systems", *Proc. Winter Simulation Conference*, 1986, pp. 52-58.

[13] P.W.Glynn, "Stochastic Approximation for Monte Carlo Optimization", *Winter Simulation Conference*, 1986, pp. 356-364.

[14] B.Hajek, "Stochastic Approximation Methods for Decentralized Control of Multiaccess Communications", *IEEE Trans. on Communications*, pp. 176-184, Vol. IT-31, No.2, March 1985.

[15] P.Heidelberger and D.Towsley, "Sensitivity Analysis from Sample Paths Using Likelihoods", *COINS Technical Report TR87-131*, University of Massachusetts, Amherst, MA, 1987.

[16] Y.C.Ho, X.R.Cao and C.Cassandras, "Finite and Infinitesimal Perturbation Analysis of Queueing Networks", *Automatica*, 19, (4), pp. 439-445.

[17] Y.C.Ho and C.Cassandras, "A New Approach to the Analysis of Discrete Event Systems", *Automatica*, Vol. 19, No. 2, pp. 149-167.

[18] Y.C.Ho and X.Cao, "Perturbation Analysis and Optimization of Queueing Networks", *J. Optimization Theory & Applications*, Vol. 40, No. 4, Aug 1983, pp. 559-582.

[19] R.L.Kasyap, C.C.Blaydon and K.S.Fu, "Stochastic Approximation", in J.M.Mendel and K.S.Fu, editors, *Adaptive, Learning and Pattern Recognition Systems: Theory and Applications*, Academic Press, New York, 1970.

[20] L.Kleinrock, "Queueing Systems, Vol. I: Theory", *Wiley*, 1975.

[21] H.Kobayashi, "Modeling and Analysis: An Introduction to System Performance and Evaluation Methodology", *Addison-Wesley*, Reading, Mass., 1978, Chapter 4.

[22] A.Kumar and F.Bonomi, "Adaptive Load Balancing in a Multiprocessor System with a Central Job Scheduler", *Proc. 2nd Int. Workshop on Applied Mathematics and Performance/Reliability Models of Computer Communication Systems*, Univ. of Rome II, pp. 173-188, 1987.

[23] H.J.Kushner and D.S.Clark, "Stochastic Approximation Methods for Constrained and Unconstrained Systems," *Springer Verlag*, New York, 1978.

[24] H.J.Kushner, "Approximations and Weak Convergence Methods for Random Processes", *M.I.T. Press*, Cambridge, 1984.

[25] S.Lakshmivarahan, "Learning Algorithms: Theory and Applications", *Springer-Verlag*, New York.

[26] K.J.Lee, "Load Balancing in Distributed Computer Systems", Ph.D. Dissertation, Univ. of Massachusetts, Amherst, 1987.

[27] Y.T.Leung and R.Suri, "Convergence of a Single Run Simulation Optimization Algorithm", *Proc. American Control Conf.*, 1988,

[28] L.Ljung, "Analysis of Recursive Stochastic Algorithms," *IEEE Trans. on Automatic Control*, Vol AC-20, 1977, pp. 551-575.

[29] D.G.Luenberger, "Linear and Nonlinear Programming", *Addison-Wesley*, Reading, MA, 1984.

[30] M.S.Meketon, "Optimization in Simulation: A Survey of Recent Results", *Proc. Winter Simulation Conf.*, 1987.

[31] S.Pulidas, D.Towsley and J.Stankovic, "Design of Efficient Parameter Estimators for Decentralized Load Balancing Policies", *Proc. 8th Int. Conf. on Distributed Computing Systems*, San Jose, 1988.

[32] M.I.Reiman and A.Weiss, "Sensitivity Analysis for Simulations Via Likelihood Ratios", *Operations Research*, to appear.

[33] H.Robbins and S.Munro, "A Stochastic Approximation Method," *Ann. Math. Stat.*, Vol 22, 1951, pp. 400-407

[34] V.K.Rohatgi, "An Introduction to Probability Theory and Mathematical Statistics", *Wiley*, 1976.

[35] D.Russell, “Optimization Theory,” *W.A.Benjamin Inc.*, New York, 1970.

[36] R.Simha and J.F.Kurose, “Stochastic Approximation Schemes for a Load Balancing Problem”, *COINS Technical Report TR 89-10*, Computer and Information Sciences Department, University of Massachusetts, Amherst, MA, 1989.

[37] R.Simha and J.F.Kurose, “Stochastic Approximation Schemes for a Resource Allocation Problem with Unknown Constraints”, *COINS Technical Report TR 89-11*, Computer and Information Sciences Department, University of Massachusetts, Amherst, MA, 1989.

[38] R.Simha and J.F.Kurose, “Relative Reward Strength Algorithms for Learning Automata”, *IEEE Trans. on Systems, Man & Cybernetics*, to appear.

[39] T.E.Stern, “A Class of Decentralized Routing Algorithms Using Relaxation”, *IEEE Trans. on Communications*, pp. 1092-1102, Vol. COM-25, No. 10, Oct 1977.

[40] R.Suri and M.A.Zazanis, “Perturbation Analysis Gives Strongly Consistent Estimates for the M/G/1 Queue”, *Technical Report*, Division of Applied Sciences, Harvard University, Cambridge, MA.

[41] R.Suri, “Perturbation Analysis: The State of the Art and Research Issues Explained Via the GI/G/1 Queue”, *Proc. IEEE*, Vol. 77, No. 1, Jan 1989, pp. 114-137.

[42] J.N.Tsitsiklis and D.P.Bertsekas, “Distributed Asynchronous Optimal Routing in Data Networks,” *Proc. of the 23rd Conf. on Decision & Control*, 1984.

[43] S.R.S.Varadhan, Large Deviations and Applications, *SIAM*, Philadelphia, 1984.

[44] Y.Wardi, “Simulation-based Stochastic Algorithms for Optimizing GI/G/1 Queues”, manuscript, Ben Gurion University of the Negev, 1988.

[45] M.T.Wasan, “Stochastic Approximation”, *Cambridge Univ. Press*, 1969.

[46] R.J.Williams, “Reinforcement Learning in Connectionist Networks: A Mathematical Analysis,” Report 8605, *Institute for Cognitive Science*, University of California, San Diego, 1986.

Appendix:

PROOF OF THEOREM 1: Let

$$J(\alpha) = \sup_{X \in D} \log E \left[\exp \left(\sum_{i=1}^{p} \alpha_i \left(\hat{Y}_i^j(X) - Y_i(X) \right) \right) \right].$$

The function J is convex [9] and by assumption, $\frac{\partial J(\alpha)}{\partial \alpha}$ exists at $\alpha = 0$; this can be computed and found to be $\frac{\partial J(\alpha)}{\partial \alpha} = 0$. Define L as the Legendre-Fenchel transform [9] of J:

$$L(\beta) = \sup_{\alpha} [< \alpha, \beta > - J(\alpha)] .$$

Now, $J(0) = 0$ and $\frac{\partial J(\alpha)}{\partial \alpha} = 0$ imply $L(\beta) = 0$ iff $\beta = 0$. The fact that J is finite in a neighbourhood of $\alpha = 0$ and that J is convex together imply $L(\beta) \to \infty$ as $|\beta| \to \infty$ [9]. We have therefore shown

(i') $L(\beta) = 0$ iff $\beta = 0$.

(ii') The sets $\{\beta : L(\beta) \leq s\}$ are compact for all $s \in [0, \infty)$.

By Theorem $II.2$ in [9], we obtain

(iii') Given $\delta > 0$ and $s > 0$ there exists $M < \infty$ such that uniformly in $X \in D$ and A satisfying $\inf_{\beta \in A} L(\beta) \geq s$,

$$P\left[\left(\tilde{Y}_1^m(X), ..., \tilde{Y}_p^m(X)\right) \in A\right] \leq \exp\left[-(s - \delta) m\right]$$

for all $m \geq M$. This implies Theorem 1 if we define $L_Y(\beta) = L\left(\beta - (Y_1(X), ..., Y_p(X))\right)$. □

PROOF OF THEOREM 2: Theorem 2 follows from Theorem 1 via the upper bound part of the 'contraction principle' (see pages 5 and 6 in [43]). □

PROOF OF THEOREM 3: We will prove that $P\left[\left|X^{(n)} - X^*\right| > \delta\ i.o.\right] = 0$ for all $\delta > 0$ (here, the notation $i.o.$ refers to 'infinitely often' [34]). Fix $\delta > 0$. Define

$$E_i = \left\{\left|X^{(n)} - X^*\right| > \delta \text{ for some } n,\ iN \leq n \leq iN + N\right\}$$

where N is from assumption $A3$ for our chosen value of δ. Clearly,

$$\left\{\left|X^{(n)} - X^*\right|\ i.o.\right\} = \{E_i\ i.o.\}\quad a.s. \tag{16}$$

where the first event in equation (16) is $i.o.$ in n whereas the second is $i.o.$ in i. Next, the stability assumption, $A3$, implies that uniformly in $X \in D$, with ϵ from $A3$,

$$\begin{aligned} P\left[E_i \middle| X^{iN-N} = x\right] \quad \leq \quad P\Bigg[\sup_{1 \leq i \leq K} &\left|g_i\left(\tilde{Y}_1^{L(n)}(X^{(n)}), ..., \tilde{Y}_p^{L(n)}(X^{(n)})\right) - h(X^{(n)})\right| > \epsilon \\ &\text{for some } n,\ iN - N \leq n \leq iN + N \,\Big|\, X^{iN-N} = x\Bigg] \end{aligned}$$

$$\leq \sum_{n=iN-N+1}^{iN+N} P\left[\sup_{1\leq i\leq K}\left|g_i\left(\tilde{Y}_1^{L(n)}(X^{(n)}),\ldots,\tilde{Y}_p^{L(n)}(X^{(n)})\right)-h(X^{(n)})\right|>\epsilon\,\middle|\,X^{iN-N}=x\right]$$

But Theorem 2 provides an upper bound on each term in the summation that is uniform in X. Hence

$$\begin{aligned}
P\left[E_i\right] &\leq \sum_{n=iN-N+1}^{iN+N} \exp\left(-c\left[L(n)\right]\right)\\
&\leq \sum_{n=iN-N+1}^{iN+N} \exp\left(-c\left[\inf_{iN-N\leq n\leq iN+N} L(n)\right]\right)\\
&\leq K(2N+1)\exp\left(-c\left[\inf_{iN-N\leq n\leq iN+N} L(n)\right]\right)
\end{aligned}$$

for i sufficiently large, where

$$c=\frac{1}{2}\inf_{|\beta-h(X)|>\epsilon} L_g(\beta)>0$$

We have broken up the sequence of iterations into blocks of size N and placed a bound on the probability of E_i via the Large Deviation result, Theorem 2. Now, we return to equation (16) in order to study the probability of E_i occuring infinitely often. If $\sum_{i=0}^{\infty} P\left[E_i\right]<\infty$ then, by the Borel-Cantelli lemma [34], this implies that $P\left[E_i \; i.o.\right]=0$. Now,

$$\begin{aligned}
\sum_{i=0}^{\infty} P\left[E_i\right] &\leq \sum_{i=0}^{\infty} K(2N+1)\exp\left(-c\left[\inf_{iN-N\leq n\leq iN+N} L(n)\right]\right)\\
&\leq K(2N+1)\sum_{n=0}^{\infty}\exp\left(-c\left[L(n)\right]\right)\\
&= K(2N+1)\sum_{n=0}^{\infty} n^{-c[c_n]}
\end{aligned}$$

which is a finite sum since the product $c[c_n]>2$ for large enough n and hence convergence follows. □