Математички Билтен
41(LXVII) No. 1
2017(62-79)
Скопје, Македонија

ISSN 0351-336X (print)
ISSN 1857-9914 (online)
UDC: 519.856:519.244

STOCHASTIC APPROXIMATION WITH ADAPTIVE STEP SIZES FOR OPTIMIZATION IN NOISY ENVIRONMENT AND ITS APPLICATION IN REGRESSION MODELS

MILENA KRESOJA, MARKO DIMOVSKI, IRENA STOJKOVSKA,
AND ZORANA LUŽANIN

Abstract. We propose a generalization of recently proposed stochastic approximation method with adaptive step sizes for optimization problems in noisy environment. The adaptive step size scheme uses only a predefined number of last noisy functional values to select a step size for the next iterate and allows different intensities of influence of the past functional values. The almost sure convergence is established under suitable assumptions. Numerical results indicate a good performance of the method. Application of the method in regression models is presented.

1. Introduction

In this paper we consider the following optimization problem in noisy environment

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable, possibly nonconvex function bounded below on $\mathbb{R}^n$. We assume that only noisy measurements of the objective function $f(x)$ and its gradient $g(x)$ are available at every $x \in \mathbb{R}^n$ i.e.

$$F(x) = f(x) + \xi \quad \text{and} \quad G(x) = g(x) + \varepsilon, \tag{2}$$

where ξ and ε represent the noise terms, random variable and random vector defined on a probability space $(\Omega, \mathcal{F}, P)$. Moreover, we assume that there is a unique solution $x^* \in \mathbb{R}^n$ of problem (1). We will use the following notation

$$\begin{aligned} F_k = F_k(x_k) = f(x_k) + \xi_k = f_k + \xi_k \\ G_k = G_k(x_k) = g(x_k) + \varepsilon_k = g_k + \varepsilon_k, \end{aligned} \tag{3}$$

2000 Mathematics Subject Classification. 90C15, 62L20, 65K05.

Key words and phrases. unconstrained optimization, stochastic optimization, stochastic approximation, noisy function, adaptive step size, gradient method, descent direction, regression models.

where the index k used with ε and ξ allows us to consider the case when the noise depends on the current iterate x_k.

The most common approach for solving the problem (1) is *Stochastic Approximation* (SA) algorithm [13]. For a given initial approximation, iterative rule of SA algorithm is given by formula

$$x_{k+1} = x_k - a_k G_k,\ k = 0, 1, 2, ... \tag{4}$$

where a_k is a nonnegative step size and G_k is the noisy measurement of the gradient at a current iterate x_k. It mimics deterministic descent direction method and uses only noisy gradient measurements. The mean square (m.s.) convergence is established by Robbins and Monro, [13], while the almost sure (a.s.) convergence is proved by Chen [2] and Spall [16]. Iterative rule of the SA algorithm (4) depends heavily on the step size sequence $\{a_k\}$ which determines the rate of convergence. The most used step size sequence is

$$a_k = \frac{a}{(k+1+A)^\alpha}, \tag{5}$$

where $a > 0$, $A \geq 0$ and $0.5 < \alpha \leq 1$. However, the step sizes (5) are proportional to $1/k$ which results in a quite slow progress. The step size selection is discussed in many papers ([3, 5, 9, 15, 16, 22, 24]). There are also many modifications of the SA algorithm based on the search direction ([1, 15, 20, 21, 23]). Combined algorithms which use benefits from line search methods and stochastic approximation are proposed in [7, 8].

The conditions on the step sizes a_k which ensure convergence of the SA algorithm (4) are the following

$$a_k > 0,\ \sum_k a_k = \infty \text{ and } \sum_k a_k^2 < \infty. \tag{6}$$

The conditions (6) provide that the step size a_k doesn't decay too fast neither too slow. These conditions are the most relevant from user's point of view. We will state the rest of convergence conditions.

Let $\{x_k\}$ be a sequence generated by SA method (4) and let $\mathcal{F}_k$ be the σ-algebra generated by $x_0, x_1, \ldots, x_k$. The set of standard assumptions consists of the three following assumptions, [2]:

A1 For any $\delta > 0$ there is $\beta_\delta > 0$ such that

$$\inf_{||x-x^*||>\delta} (x-x^*)^T g(x) = \beta_\delta > 0.$$

A2 The observation noise $(\varepsilon_k, \mathcal{F}_{k+1})$ is a martingale difference sequence with

$$E(\varepsilon_k|\mathcal{F}_k) = 0 \text{ and } E[||\varepsilon_k||^2] < \infty \text{ a.s for all } k,$$

where $\{\mathcal{F}_k\}$ is a family of nondecreasing σ-algebras.

A3 The gradient g and the conditional second moment of the observation noise have the following upper bound

$$||g(x)||^2 + E(||\varepsilon_k||^2|\mathcal{F}_k) < c(1 + ||x - x^*||^2) \text{ a.s. for all } k \text{ and } x \in \mathbb{R}^n,$$

where $c > 0$ is a constant.

Assumption A1 is condition on the shape of $g(x)$, assumption A2 is the mean-zero noise condition, while assumption A3 is a restriction on the magnitude of $g(x)$.

The following theorem gives a convergence result for SA method (4).

Theorem 1. [2] *Assume that A1-A3 hold. Let $\{x_k\}$ be a sequence generated by SA method (4) with the gain sequence $\{a_k\}$ satisfying (6). Then the sequence $\{x_k\}$ converges to x^* for an arbitrary initial approximation x_0.*

Now, we will review a case when SA method (4) uses a descent direction instead of a negative gradient. Direction d_k is a *descent direction* at x_k if

$$G_k^T d_k < 0, \tag{7}$$

where G_k is the noisy gradient at x_k (see [7] for details). For a given initial approximation x_0, the iterative rule for the SA method with descent direction is given by the formula

$$x_{k+1} = x_k + a_k d_k, \tag{8}$$

where a_k is a nonnegative step size and d_k is a descent direction defined by (7). The set of assumptions which ensures the convergence of (8) is similar to one needed for convergence of (4).

Let $\{x_k\}$ be a sequence generated by (8) and $\mathcal{F}_k$ is the σ-algebra generated by $x_0, x_1, \ldots, x_k$. Two additional assumptions that are imposed are, [7]:

A4 There exist $c_1 > 0$ such that direction d_k satisfies

$$(x_k - x^*)^T E(d_k|\mathcal{F}_k) \leq -c_1||x_k - x^*|| \quad \text{a.s. for all } k.$$

A5 There is $c_2 > 0$ such that

$$||d_k|| \leq c_2||G_k|| \quad \text{a.s. for all } k.$$

The assumption A4 limits the influence of noise on d_k, while the assumption A5 makes connection of the available noisy gradient with descent direction.

Theorem 2. [7] *Assume that A2-A5 hold. Let $\{x_k\}$ be a sequence generated by* (8) *with the gain sequence $\{a_k\}$ satisfying (6). Then the sequence $\{x_k\}$ converges to x^* a.s. for an arbitrary initial approximation x_0.*

Recently proposed SA algorithm with adaptive step sizes uses a general descent direction d_k defined by (7), and finds the next iterate according to the iterative rule (8), where step sizes a_k are defined by, [9]:

$$a_k = \begin{cases} a\theta^{s_k}, & F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \sigma \\ 0, & F_k > \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \sigma, \\ \frac{a}{(t_k+1+A)^\alpha}, & otherwise \end{cases} \tag{9}$$

where $m(k) = \min\{k, m\}$, $m \in \mathbb{N}$, $\theta \in (0,1)$, $a > 0$, $A \geq 0$, $0.5 < \alpha \leq 1$, $\sigma > 0$, s_k counts the occurrences of the events, $\left\{F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \sigma\right\}$, and t_k counts the occurrences of the events $\left\{\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \sigma \leq F_k \leq \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \sigma\right\}$. The adaptive step size rule (9) tracks the previous observed function values F_{k-1}, F_{k-2}, ..., $F_{k-m(k)}$, to get insight into whether the objective function is improving. If there is a "sufficient" decrease in the objective function, a larger step size $a_k = a\theta^{s_k}$ is used. Bed steps are blocked using zero step size. Otherwise, a backup step size $a_k = \frac{a}{(t_k+1+A)^\alpha}$ is used. At each iteration, an interval

$$J_k = (\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \sigma, \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \sigma)$$

is constructed, that acts as a (hybrid) interval for the expected optimal function value f^*, since it is symmetrical about the sample mean i.e. around $\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j}$. And if the next estimate F_k of f^* is in the interval J_k, a slow but safe steps are used. In [9], it is shown that under reasonable assumptions on the noise terms, the step sizes defined by (9) satisfy the conditions (6) a.s. This result and the SA convergence theorems, Theorem 1 and Theorem 2, adapted for stochastic step sizes, ensure almost sure convergence of the SA algorithm with adaptive step sizes (9), see [9].

In this paper we propose a generalization of the step size scheme (9) and a corresponding adaptive step size algorithm. The proposed generalized step size scheme allows past functional values to have not equal influence while selecting a new step size length, and allows constructing bigger steps when a "sufficient" decrease in the objective function is monitored. Almost sure convergence of the proposed algorithm is established and the algorithm is tested on a set of standard test problems. Application to regression models of the proposed method is explored.

The organization of the paper is the following. A new generalized step size scheme and the algorithm are presented in Section 2. Convergence theory is given in the same section. Numerical results are given in Section 3, while application of the proposed method in regression models is presented in Section 4. Conclusions are drawn in Section 5.

2. Algorithm and Convergence Results

2.1. **The Step Size Scheme and the Algorithm.** We propose a generalization of the step size scheme (9) which allows to use information from previous steps in a more general way and to define bigger step sizes when a "sufficient" decrease in the objective function is monitored.

Let us assume that the index of the current iterate is k and we wish to determine the step size a_k in the next iterate. Denote by $\sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j}$ a convex combination of $m(k)$ previous noisy function values F_{k-1}, F_{k-2}, ..., $F_{k-m(k)}$, where $m(k) = \min\{k, m\}$, $m \in \mathbb{N}$ and $\lambda_{k,j} \geq \lambda > 0$, $j = 1, 2, \ldots, m(k)$ such that $\sum_{j=1}^{m(k)} \lambda_{k,j} = 1$, for all k. Now, we will consider the following interval

$$\tilde{J}_k = (\sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} - \sigma, \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} + \sigma), \tag{10}$$

where $\sigma > 0$. If a noisy function value in kth iteration, F_k, is lower than the lower limit of the interval $\tilde{J}_k$, we will declare progress of the algorithm and use a larger step in the next $(k+1)$th iterate. If F_k is greater than the upper limit of the interval $\tilde{J}_k$, we will declare iteration as a bad step and put $x_{k+1} = x_k$. If F_k lies in the interval $\tilde{J}_k$, the step size similar to (5) is taken. Detail formulation of our adaptive step size scheme is given by:

$$a_k = \begin{cases} b\theta^{s_k}, & F_k < \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} - \sigma \\ 0, & F_k > \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} + \sigma, \\ \frac{a}{(t_k+1+A)^\alpha}, & otherwise \end{cases} \tag{11}$$

where

- $m(k) = \min\{k, m\}$, $m \in \mathbb{N}$, $\sigma > 0$, $\theta \in (0, 1)$, $b \geq a > 0$, $A \geq 0$, $0.5 < \alpha \leq 1$,
- $\lambda_{k,j} \geq \lambda \geq 0$, $j = 1, \ldots, m(k)$ such that $\sum_{j=1}^{m(k)} \lambda_{k,j} = 1$,
- $s_k = s_{k-1} + I\left\{F_k < \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} - \sigma\right\}$, for $k = 1, 2, ...,$ and $s_0 = 0$,
- $t_k = t_{k-1} + I\left\{\sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} - \sigma \leq F_k \leq \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} + \sigma\right\}$, for $k = 1, 2, ...,$ and $t_0 = 0$,

where $I(\cdot)$ stands for the indicator function.

Adaptive step sizes (11) differs from (9) in the expression $\sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j}$ which allows previous functional values to be taken with different intensities at each iteration k then in the arithmetic mean $\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j}$. In this way, the step size scheme (11) can use more effectively the information about the optimization process stored in previous function values. Another advantage is that the bigger step sizes, when a "sufficient" decrease in the objective function is monitored, can be taken, because of the parameter b in the step size $b\theta^{s_k}$ that is allowed to be larger than the parameter a in the step size $\frac{a}{(t_k+1+A)^\alpha}$. The step size scheme (9) is a special case of the step size scheme (11), where $\lambda_{k,j} = \frac{1}{m(k)}$, for all $j = 1, 2, ..., m(k)$.

Finally we can formulate the algorithm with adaptive step size selection scheme (11).

Algorithm 1. *CC-Adaptive Stochastic Approximation Algorithm*

Step 0. Initialization. Choose an initial point $x_0 \in \mathbb{R}$, constants $\sigma > 0$, $m \in \mathbb{N}$, $\theta \in (0, 1)$, $b \geq a > 0$, $A \geq 0$, $0.5 < \alpha \leq 1$ and $\lambda > 0$. Set $k = 0$.

Step 1. Choose $\lambda_{k,j} \geq \lambda > 0$, $j = 1, \ldots, m(k)$ such that $\sum_{j=1}^{m(k)} \lambda_{k,j} = 1$.

Step 2. Direction selection. Choose d_k such that (7) holds.

Step 3. Step size selection. Calculate the noisy function measurement F_k and select the step size a_k according to the criterion (11).

Step 4. Update iteration. Calculate $x_{k+1} = x_k + a_k d_k$, set $k = k+1$ and go to Step 1.

Note that the Algorithm 1 is formulated for an arbitrary constant $\sigma > 0$. In our numerical experiments, Section 3, we have chosen σ to be equal to the noise level, since the mean-square error (MSE) of the function estimator F_k of the optimal value f^*, which is equal to $\sigma^2 + (f_k - f^*)^2$, is often a reasonably good approximation for the variance of the sampling distribution of F_k, [6].

2.2. **Properties of the Adaptive Step Size Scheme and Convergence of the Algorithm.** In this subsection, we will show that the sequence $\{a_k\}$ generated by (11) satisfies the conditions (6) a.s. under reasonable assumption on the noise terms ξ_k. Namely, we will suppose that

$$\begin{aligned}&\xi_k, k = 0, 1, 2, ... \text{ are i.i.d. continuous random variables with common}\\ &\text{probability density function (pdf) } p(x) > 0 \text{ a.s. for all } x \in \mathbb{R}.\end{aligned} \tag{12}$$

The condition (12) is often satisfied in practice since the noise usually occurs independently. The independent distributed normal Gaussian noise is the one that satisfies conditions (12).

Denote by

$$A_k = \left\{a_{k-1} = a_{k-2} = \ldots = a_{k-m(k)} = 0\right\}, \tag{13}$$

the event that $m(k)$ consecutive zero steps have occurred.

Lemma 1. *Let the step sizes a_k be defined by (11). If the noise terms ξ_k satisfy the conditions (12), then for $k = 1, 2, \ldots$, the following inequality holds*

$$P(A_k) > 0, \tag{14}$$

where A_k is the event defined by (13).

Proof. Let us assume the contrary, that there exists k such that

$$0 = P(A_k) = P(F_{k-i} > \sum_{j=1}^{m(k)} \lambda_{k-i,j} F_{k-i-j} + \sigma,\ i = 1, 2, \ldots, m(k)). \tag{15}$$

Since for each k, $\sum_{j=1}^{m(k)} \lambda_{k,j} = 1$, we have

$$\left\{F_{k-i} > \max_{1\leq j\leq m(k)} F_{k-i-j} + \sigma\right\} \subseteq \left\{F_{k-i} > \sum_{j=1}^{m(k)} \lambda_{k-i,j} F_{k-i-j} + \sigma\right\},$$

for each $i = 1, 2, ..., m(k)$, so

$$\bigcap_{i=1}^{m(k)} \left\{F_{k-i} > \max_{1\leq j\leq m(k)} F_{k-i-j} + \sigma\right\} \subseteq \bigcap_{i=1}^{m(k)} \left\{F_{k-i} > \sum_{j=1}^{m(k)} \lambda_{k-i,j} F_{k-i-j} + \sigma\right\},$$

which implies

$$\begin{aligned} &P(F_{k-i} > \max_{1\leq j\leq m(k)} F_{k-i-j} + \sigma,\ i = 1, 2, ..., m(k)) \\ &\leq P(F_{k-i} > \sum_{j=1}^{m(k)} \lambda_{k-i,j} F_{k-i-j} + \sigma,\ i = 1, 2, ..., m(k)). \end{aligned} \tag{16}$$

Now, (15) and (16) imply

$$P(F_{k-i} > \max_{1\leq j\leq m(k)} F_{k-i-j} + \sigma,\ i = 1, 2, \ldots, m(k)) = 0. \tag{17}$$

Further, the proof proceeds as the proof of Lemma 3.1 in [9], and leads to a contradiction, which implies that $P(A_k) > 0$ for all k. □

Lemma 2. *Let the step sizes a_k be defined by (11). If the noise terms ξ_k satisfy the conditions (12), then for all $k = 1, 2, \ldots$*

$$P(a_k = 0|A_k) > 0, \tag{18}$$

$$P(a_k = a\theta^{s_k}|A_k) > 0, \tag{19}$$

and

$$P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}|A_k) > 0, \tag{20}$$

where A_k is the event defined by (13). Moreover, for all $k = 1, 2, ...$

$$P(a_k = 0) > 0, \tag{21}$$

$$P(a_k = a\theta^{s_k}) > 0, \tag{22}$$

and

$$P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}) > 0. \tag{23}$$

Proof. The conditional probabilities are well defined because of Lemma 1. Under the realization of the event A_k we have that $f_k = \sum_{j=1}^{m(k)} \lambda_{k,j} f_{k-j}$. Using the formulation of the step size rule (11), we have

$$\begin{aligned} P(a_k = 0|A_k) &= P(F_k > \sum_{j=1}^{m(k)} \lambda_{k,j} F_{k-j} + \sigma|A_k) \\ &= P(f_k + \xi_k > \sum_{j=1}^{m(k)} \lambda_{k,j}(f_{k-j} + \xi_{k-j}) + \sigma|A_k) \\ &= P(\xi_k - \sum_{j=1}^{m(k)} \lambda_{k,j}\xi_{k-j} > \sigma), \end{aligned} \tag{24}$$

since the last conditional probability is independent of the condition.

Using convolution formula for independent random variables it can be easily proved that the random variable $\xi_k - \sum_{j=1}^{m(k)} \lambda_{k,j}\xi_{k-j}$ has positive density function. Therefore,

$$P(a_k = 0|A_k) = P(\xi_k - \sum_{j=1}^{m(k)} \lambda_{k,j}\xi_{k-j} > \sigma) > 0. \tag{25}$$

Similarly,

$$P(a_k = a\theta^{s_k}|A_k) = P(\xi_k - \sum_{j=1}^{m(k)} \lambda_{k,j}\xi_{k-j} < -\sigma) > 0 \tag{26}$$

and

$$P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}|A_k) = P(-\sigma \leq \xi_k - \sum_{j=1}^{m(k)} \lambda_{k,j}\xi_{k-j} \leq \sigma) > 0, \tag{27}$$

since $\sigma > 0$. Relations (21)-(23) are a direct consequence from Lemma 1 and (25)-(26), see Lemma 3.2 from [9] for details, which completes the proof. □

Lemma 2 leads to important result which is stated below.

Lemma 3. *Let the step sizes a_k be defined by (11). If the noise terms ξ_k satisfy the condition (12), then almost surely there are infinitely many steps $a_k = \frac{a}{(t_k+1+A)^\alpha}$ and infinitely many steps $a_k = b\theta^{s_k}$.*

Proof. Same as the proof of Lemma 3.3 in [9]. □

We will state now the most important property of the step size sequence $\{a_k\}$ defined by (11).

Theorem 3. *If the noise terms ξ_k satisfy the condition (12), then the step size sequence $\{a_k\}$, defined by (11), satisfies the conditions (6) a.s.*

Proof. Same as the proof of Theorem 3.1 in [9]. □

Previously mentioned SA convergence theorems, Theorem 1 and Theorem 2, assume deterministic step sizes a_k that satisfy conditions (6). In order to use these results when step sizes a_k are stochastic, we need to assume that a_k is $\mathcal{F}_k$-measurable, where $\mathcal{F}_k$ is the σ-algebra generated by $x_0, x_1, x_2, ..., x_k$, and $\{x_k\}$ is a sequence generated by the corresponding algorithm, similarly as it is assumed in [9, 11]. When step sizes a_k are stochastic, we also need to assume that conditions (6) are satisfied almost surely (a.s.). Under those additional assumptions, SA convergence theorems, Theorem 1 and Theorem 2, also hold when step sizes a_k are stochastic.

Now, for the step sizes a_k generated by Algorithm 1, Theorem 3 ensures almost surely fulfilment of the conditions (6), which together with assumptions A2-A5 ensures almost surely convergence of Algorithm 1, due to the convergence theorem for descent direction method with SA step sizes, Theorem 2 for stochastic step sizes a_k. So, we have the following convergence result for the method with adaptive step sizes proposed in Algorithm 1.

Theorem 4. *Assume that A2-A5 hold. Let $\{x_k\}$ be a sequence generated by Algorithm 1, where the noise terms ξ_k satisfy the condition (12). Then the sequence $\{x_k\}$ converges to x^* a.s. for an arbitrary initial approximation x_0.*

The convergence of Algorithm 1 with $d_k = -G_k$, is a direct consequence of SA convergence theorem, Theorem 1 for stochastic step sizes a_k, and the property of the gain sequence $\{a_k\}$ given with Theorem 3.

Corollary 1. *Assume that A1-A3 hold. Let $\{x_k\}$ be a sequence generated by Algorithm 1 with $d_k = -G_k$, where the noise terms ξ_k satisfy the condition (12). Then the sequence $\{x_k\}$ converges to x^* a.s. for an arbitrary initial approximation x_0.*

3. Numerical Results

Algorithm 1 is tested with $d_k = -G_k$ and BFGS direction $d_k = -B_k^{-1} G_k$, with the update formula

$$B_{k+1} = B_k - \frac{B_k \delta_k \delta_k^T B_k}{\delta_k^T B_k \delta_k} + \frac{\Delta_k \Delta_k^T}{\Delta_k \delta_k}, \tag{28}$$

where

$$\delta_k = x_{k+1} - x_k \quad \text{and} \quad \Delta_k = G(x_{k+1}, \varepsilon_k) - G(x_k, \varepsilon_k),$$

i.e. the gradient difference Δ_k is calculated using the same sample set which is already successfully tested in [7, 14, 19].

We have chosen 18 test problems from [10] and [12]. The problems are listed in Table 1. The normal distributed noise was added to the function and gradient evaluations to transform original problems into problems in noisy environment i.e. the form of the noise is

$$\xi \sim \mathcal{N}(0,\sigma^2) \quad \text{and} \quad \varepsilon \sim \mathcal{N}(0,\sigma^2 I_{n\times n}),$$

where σ represent the noise level and $I_{n\times n}$ is the identity matrix. Two different levels of the noise, $\sigma = 0.4, 1$ a are tested. The noisy function and gradient are calculated using the arithmetic mean with sample size $p = 3$ i.e.

$$F_k = \frac{1}{p}\sum_{i=1}^{p} F(x_k, \xi_k^i) \quad G_k = \nabla F(x_k, \varepsilon_k^i),$$

where $\left\{\xi_k^i\right\}$ and $\left\{\varepsilon_k^i\right\}$ are i.i.d. samples.

Each test had $N = 50$ independent runs starting from the same initial point. The runs are grouped in three categories: successful (convergent), partially successful runs and unsuccessful (divergent) runs. Run is successful if a method stops due to $||G_k|| \le c = \min\{\sqrt{n}\sigma, 1\}$. The number of successful runs is denoted by $Nconv$. If $||G_k|| > 200\sqrt{n}$, run is unsuccessful. The number of divergent runs is denoted by $Ndiv$. If the runs stops due to reaching maximal number of $200n$ function evaluations are considered partially successful and their number is denoted by $Npar$.

The specification of the scheme (11) is as follows. The values of parameters a, A and α are given in Table 2, while the value of parameter b is specified within the algorithms that we compare. Results that we present are for $m = 10$ and $\theta = 0.99$. Two sets of coefficients $\lambda_{k,j}$ are considered. The first case is when arithmetic mean is used i.e. for all k, $\lambda_{k,j} = \frac{1}{m(k)}$, $j = 1, \ldots, m(k)$. The second case is when the coefficients $\lambda_{k,j}$ are chosen as following:

$$\lambda_{k,1} = \begin{cases} 1, & F_k > \sum_{j=1}^{m(k)} \tilde{\lambda}_{k,j} F_{k-j} \\ \tilde{\lambda}_{k,1}, & \text{otherwise} \end{cases}, \tag{29}$$

and

$$\lambda_{k,j} = \begin{cases} 0, & F_k > \sum_{j=1}^{m(k)} \tilde{\lambda}_{k,j} F_{k-j} \\ \tilde{\lambda}_{k,j}, & \text{otherwise} \end{cases}, \; j = 2, ..., m(k), \tag{30}$$

where

$$\tilde{\lambda}_{k,j} = \lambda, \; j \ne \tilde{j} \text{ and } \tilde{\lambda}_{k,\tilde{j}} = 1 - (m(k) - 1)\lambda, \tag{31}$$

where $\lambda = 0.01$ and $\tilde{j}$ is such that $F_{k-\tilde{j}} = \max_{1\le j\le m(k)} F_{k-j}$.

As a consequence of Lemma 3, we can limit the number of consecutive zero steps, since almost surely infinitely many consecutive zero steps can not occur. If the number of consecutive zero steps is greater than $m + 1$,

Problem	n	x_0
The Gaussian function	3	$(4/10, 1, 0)$
The Box 3-dimensional function	3	$(0, 10, 5)$
The variably dimensioned function	4	$(3/4, 2/4, 1/4, 0)$
The Watson function	4	$(0, 0, 0, 0)$
The Penalty Function 1	10	$(1, 1, \ldots, 1)$
The Penalty Function 2	4	$(1/2, 1/2, 1/2, 1/2)$
The Trigonometric Function	10	$(1/10, 1/10, \ldots, 1/10)$
The Beale Function	2	$(1, 1)$
The Chebyquad Function	10	$(5/11, 10/11 \ldots, 50/11)$
The Gregory and Karney Tridiagonal Matrix Function	4	$(0, 0, 0, 0)$
The Hilbert Matrix Function	4	$(1, 1, 1, 1)$
The De Jong Function 1	3	$(-5.12, 0, 5.12)$
The Branin RCOS Function	2	$(-1, 1)$
The Colville Polynomial	4	$(1/2, 1, -1/2, -1)$
The Powell 3D Function	3	$(0, 1, 2)$
The Himmelblau function	2	$(-1.3, 2.7)$
Strictly Convex 1	10	$(1/10, 2/10, \ldots, 1)$
Strictly Convex 2	10	$(1, 1, \ldots, 1)$

Table 1. Test problems

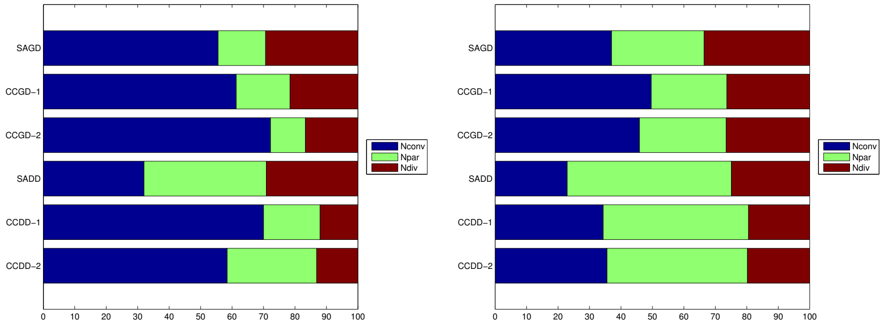
we use $a_k = \frac{a}{(t_k+1+A)^\alpha}$ in next iterate. We present results for the following 6 algorithms:

- SAGD - Algorithm (4) with SA step sizes (5)
- CCGD-1- Algorithm 1 with $b = a$, $d_k = -G_k$ and $\lambda_{k,j} = \frac{1}{m(k)}$
- CCGD-2 - Algorithm 1 with $b = 1$, $d_k = -G_k$ and $\lambda_{k,j}$ as in (29)-(31)
- SADD - Algorithm (8) $d_k = -B_k^{-1} G_k$ and SA step sizes (5)
- CCDD-1 - Algorithm 1 with $b = a$, $d_k = -B_k^{-1} G_k$ and $\lambda_{k,j} = \frac{1}{m(k)}$
- CCDD-2 - Algorithm 1 with $b = 1$, $d_k = -B_k^{-1} G_k$ and $\lambda_{k,j}$ as in (29)-(31)

Figure 1 shows the overviews of successful, partially successful and unsuccessful runs of the algorithms for both noise levels $\sigma = 0.4$ and $\sigma = 1$. The obtained results indicate that our algorithm has smaller number of divergent runs when it is compared to corresponding classical SA algorithms regardless of the noise level. For a smaller noise level and gradient direction, the algorithm that uses coefficients $\lambda_{k,j}$ of the form (29)-(31) has smaller number of divergent runs than the algorithm that uses equal coefficients $\lambda_{k,j}$.

TABLE 2. The initialization of the parameters a, A and α.

Problem	a	A	α
1	1	1	0.75
2	1	100	0.501
3	0.1	1	0.75
4	0.1	1	0.75
4	0.1	1	0.75
5	0.1	1	0.75
6	0.1	100	0.501
7	1	100	0.501
8	1	100	0.501
9	0.1	100	0.75
10	0.5	1	0.501
11	0.5	1	0.501
12	0.1	100	0.75
13	0.5	1	0.501
14	1	100	0.501
15	0.1	100	0.75
16	0.5	1	0.501
17	0.5	100	0.501
18	0.1	100	0.75



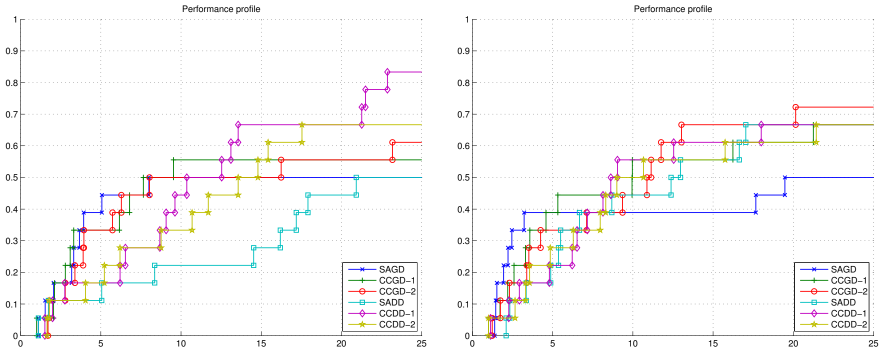
a. $\sigma = 0.4$ b. $\sigma = 1$

FIGURE 1. Percentage of successful, partially successful and divergent runs

The performance measure that we use is the number of function evaluations in successful and partially successful runs

$$\pi_{ij} = \frac{1}{|Ncon_{ij} \bigcup Npar_{ij}|} \sum_{r \in Ncon_{ij} \bigcup Npar_{ij}} \frac{fcalc^r_{ij}}{n_j},$$

where $Ncon_{ij}$ is the set of indices of successful runs for ith Algorithm to solve problem j, $Npar_{ij}$ is the set of indices of partially successful runs for ith Algorithm to solve problem j, $fcalc_{ij}^{r}$ is the number of function evaluations needed for ith Algorithm to solve problem j in rth run and n_j is the dimension of problem j. Performance profiles are given for two noise levels $\sigma = 0.4$ and $\sigma = 1$ on Figure 2.



a. $\sigma = 0.4$ b. $\sigma = 1$

FIGURE 2. Performance profiles

Performance profiles confirm the efficiency of the proposed method with adaptive step sizes, which means that SA with the adaptive step size scheme (11) overperforms classical SA with the step sizes of the form (5), regardless of the search direction, and on different noise levels. For a smaller noise level, the BFGS direction and equal coefficients $\lambda_{k,j}$ is the most robust choice, and for the larger noise level, the gradient direction and the choice (29)-(31) of the coefficients $\lambda_{k,j}$ is the most robust.

4. Application to Regression Models

In this section we will consider the linear regression model given by it's matrix form

$$y = X\beta + \epsilon, \tag{32}$$

where:

- $y = (y_1, y_2, ..., y_n)^T$ is n-vector of dependent variables,
- $X = [x_{ij}]_{n \times p}$ is $n \times p$-matrix of independent variables,
- $\beta = (\beta_1, \beta_2, ..., \beta_p)^T$ is p-vector of associated regression coefficients, and
- $\epsilon = (\epsilon_1, \epsilon_2, ..., \epsilon_n)^T$ is n-vector which components are independent and identically distributed random errors with $E(\epsilon_i) = 0$ and $D(\epsilon_i) = \sigma^2$.

The most commonly used method for estimating the unknown parameters $\beta_1, \beta_2, ..., \beta_p$ is the Ordinary Least-Square (OLS) method, where the residual square error

$$RSS = \sum_{i=1}^{n} \big(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\big)^2$$

is minimized. In other words, parameter estimates are obtained by solving the unconstrained OLS optimization problem

$$\hat{\beta}^{ols} = \arg\min_{\beta\in\mathbb{R}^p} \sum_{i=1}^{n} \big(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\big)^2. \tag{33}$$

To overcome the deficiencies of the OLS method and improve the estimates obtained by it, Tibshirani introduced the Least Absolute Shrinkage and Selection Operator (LASSO) regularization method, [17]. LASSO regularization is a process of adding constraints in the form of L_1-norm of the parameter vector β. The associated constrained optimization problem given by

$$\hat{\beta}^{lasso} = \arg\min_{\beta\in\mathbb{R}^{'}} \sum_{i=1}^{n} \big(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\big)^2 \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t, \tag{34}$$

is equivalent to the unconstrained optimization problem

$$\hat{\beta}^{lasso} = \arg\min_{\beta\in\mathbb{R}^{'}} \big\{ \sum_{i=1}^{n} \big(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\big)^2 + \mu \sum_{j=1}^{p} |\beta_j| \big\}. \tag{35}$$

Due to the nature of the L_1 penalty, the LASSO does both continuous shrinkage and automatic variable selection at the same time. The tuning parameter t controls the amount of shrinkage that is applied to the estimates. Parameters t and μ have some kind of a reciprocal relationship, [17]. In practice, the value of μ, as the level of regularization, is predefined, or it is chosen from some candidate set using selection methods as Cross-Validation, BIC or AIC.

We are going to apply SA method with adaptive step sizes given by Algorithm 1, for solving the unconstrained optimization problem (35) in order to find the estimates of the parameter vector β in the regression model (32). For the descent direction d_k in Algorithm 1, the negative noisy gradient is used i.e. $d_k = -G_k$, and results from this optimization are compared to the classical SA method (4).

Application is illustrated on the following example.

Example 1. [17] *In this example we are looking for the estimate of the parameter β in $Y = X\beta + \epsilon$, where the true value of β is*

$$\beta = (3, 1.5, 0, 0, 2, 0, 0, 0)^T.$$

We simulated $N = 50$ data sets of $n = 100$ observations, where the random errors ϵ_i, $i = 1, 2, ..., n$ are i.i.d with normal Gaussian distributions

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2), i = 1, 2, ..., n,$$

with $\sigma = 3$. The column vectors $X_i, i = 1, 2, ..., p$ of the matrix X of independent variables are chosen to have n-dimensional normal distributions

$$X_j \sim \mathcal{N}(0, C), j = 1, 2, ..., p,$$

where the covariance matrix $C = [c_{ij}]$ is such that $c_{ij} = \rho^{|i-j|}, i, j = 1, 2, ..., p$, with $\rho = 0.5$, [18]. The K-fold cross-validation with $K = 5$ is used to estimate the regularization level μ in (35), [4]. As a candidate set for the regularization parameter μ, the set $\{0, 0.01, 0.1, 1, 10, 100\}$ is considered.

Three optimization methods SAGD, CCGD-1 and CCGD-2, described in the previous section, have been tested, with a difference for CCGD-2 in the choice of b, which is here $b = a$ as in CCGD-1. The values of parameters used in step sizes (5) and in step size selection rule (11) are $a = 0.001$, $A = 0, 10, 100$, $\alpha = 0.602$, $m = 10$ and $\theta = 0.99$. The gradient of the objective function in (35) is approximated by the centered finite differences with step $h = 10^{-5}$. The estimates have been obtained using MATLAB programming software.

Comparison of optimization methods is based on the evaluation of Mean Square Error (MSE) and Median Square Error (MedianSE) defined by

$$MSE = \frac{1}{N} \sum_{k=1}^{N} (\hat{\beta}^k - \beta)^T C (\hat{\beta}^k - \beta),$$

and

$$MedianSE = Median\{(\hat{\beta}^k - \beta)^T C (\hat{\beta}^k - \beta), k = 1, 2, ..., N\},$$

respectively, where $\hat{\beta}^k$ is the kth estimate of the parameter β.

In Table 3, Table 4 and Table 5, MSE and MedianSE for different value of the parameter A and different initial iterations β_0 in the optimization processes are given.

Lower value of MSE or MedianSE indicates better optimization process. As it can be seen from the results, the proposed method in this paper, SA with adaptive step sizes (11), is more global then the classical SA method (4), since it has better performance when the optimization process starts far from the solution. Locally, SA method (4) gives better results, which depend on the choice of the parameter A in the step sizes (5). SA method (4) is very sensitive on the choice of the parameter A, which is not the case for the proposed method in this paper, it manages to overcome this difficulties. Methods with different choices of the parameters λ_{kj} in (11), CCGD-1 and CCGD-2, have been equally successful with little difference in MSEs or MedianSEs, almost always in favour of CCGD-1, except for

	MSE	MedianSE
$\beta_0 = (0,0,0,0,0,0,0,0)^T$		
SAGD	0.67713284	0.65167476
CCGD-1	0.73254119	0.65512802
CCGD-2	0.73114389	0.64433737
$\beta_0 = (10,10,10,10,10,10,10,10)^T$		
SAGD	0.81613584	0.73665411
CCGD-1	0.71485402	0.66087552
CCGD-2	0.72263938	0.66571446

Table 3. MSE and MedianSE, $A = 0$

	MSE	MedianSE
$\beta_0 = (0,0,0,0,0,0,0,0)^T$		
SAGD	0.72407352	0.70555047
CCGD-1	0.74199922 *	0.67134859 *
CCGD-2	0.73055080	0.65525906
$\beta_0 = (10,10,10,10,10,10,10,10)^T$		
SAGD	1.05999336	0.99529997
CCGD-1	0.71411317	0.66605187
CCGD-2	0.72268785	0.66862433

Table 4. MSE and MedianSE, $A = 10$, (* calculated over 48 nondivergent runs)

	MSE	MedianSE
$\beta_0 = (0,0,0,0,0,0,0,0)^T$		
SAGD	1.13762219	1.17352551
CCGD-1	0.71541790	0.64763942
CCGD-2	0.72503869	0.65551539
$\beta_0 = (10,10,10,10,10,10,10,10)^T$		
SAGD	2.21081217	2.15215925
CCGD-1	0.70413645	0.63140689
CCGD-2	0.70862254	0.63606241

Table 5. MSE and MedianSE, $A = 100$

$A = 10$ and initial point $\beta_0 = (0,0,0,0,0,0,0,0)^T$, when CCGD-1 results in 2 out of 50 divergent runs, and bigger MSE and MedianSE.

5. Conclusions

In this paper we have generalized a recently proposed adaptive step size scheme that can be used in SA iterative rule, for problems in noisy environment. Using the information from the previous steps, the scheme takes larger steps if sufficient decrease in previous noisy function values is observed, and takes zero steps if unwanted increase is observed. The step size sequence obtained by the scheme, under common assumption, satisfies almost surely the main conditions for convergence. Numerical results show that adaptive step size selection improve the optimization process and can be successfully implemented in estimating the unknown parameters in regression models.

Acknowledgements. This work is partially supported by Ministry of Education, Science and Technology Development of Serbia grant No. 174030 and Ss. Cyril and Methodius University of Skopje, Macedonia scientific research projects for 2014/2015 academic year.

References

[1] D. P. Bertsekas, J. N. Tsitsiklis, *Gradient convergence in gradient methods with errors*, SIAM J Optimiz. 10(3), (2000) 627-642

[2] Chen, H. F.: Stochastic Approximation and Its Application, Kluwer Academic Publishers, New York, (2002)

[3] Delyon, B., Juditsky, A.: Accelerated stochastic approximation, SIAM J Optimiz. 3(4) 868-881 (1993)

[4] Efron, B.: The estimation of prediction error: Covariance penalties and crossvalidation, J.Amer. Statis. Assoc. Vol.99 (2004) pp.619-642

[5] Kesten, H.: Accelerated stochastic approximation, Ann. Math. Stat. 29, 41-59 (1958)

[6] Knight, K.,: Mathematical statistics, Chapman & Hall/CRC, Boca Raton, Florida (2000)

[7] Krejić, N., Lužanin, Z., Ovcin Z., Stojkovska, I.: Descent direction method with line search for unconstrained optimization in noisy environment, Optim Methods Softw, 30(6), 1164-1184 (2015)

[8] Krejić, N., Lužanin, Z., Stojkovska, I.: A gradient method for unconstrained optimization in noisy environment, Appl. Numer. Math 70, 1-21 (2013)

[9] Kresoja, M., Lužanin, Z., Stojkovska, I.: Adaptive stochastic approximation algorithm, Technical Report (2016)

[10] Moré, J.J., Garbow, B.S., Hillstrom, K.E.: Testing unconstrained optimization software, TOMS 7(1), 17-41 (1981)

[11] Powell, W. B.: Approximate Dynamic Programming: Solving the Curses of Dimensionality, Chapter 6. Stochastic Approximation Methods, John Wiley & Sons, Inc., Hoboken, New Jersey (2007)

[12] Raydan, M.: The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, SIAM J Optimiz. 7(1), 26-33 (1997)

[13] Robbins, H., Monro. S.: A stochastic approximation method, Ann. Math. Stat. 22, 400-407 (1951)

[14] Schraudolph, N.N., Yu, J., Gnter, S.: A Stochastic Quasi-Newton Method for Online Convex Optimization, AISTA TS'07, 433-440 (2007)

[15] Spall, J. C.: Adaptive stochastic approximation by the simultaneous perturbarion method, IEEE AC 45(10), 1839-1853 (2000)

[16] Spall, J. C.: Introduction to stochastic search and optimization: estimation, simulation, and control, John Wiley & Sons, Inc., Hoboken, New Jersey, (2003)

[17] Tibshirani,R.: Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society, Series B Vol.58, (1996) pp.267-288

[18] Tibshirani,R.: The Lasso method for variable selection in the Cox model, Statistics in medicine, Vol.16, (1997) pp.385-395

[19] Wang, X., Ma, S., Liu, W.,: Stochastic quasi-Newton methods for nonconvex stochastic optimization, arXiv:1412.1196 [math.OC], (2014)

[20] Xu, Z.: A combined direction stochastic approximation algorithm, Optim. Lett. 4(1), 117-129 (2010)

[21] Xu, Z., Dai, Y.H.: A stochastic approximation frame algorithm with adaptive directions, NM-TMA 1(4), 460-474 (2008)

[22] Xu, Z., Dai, Y. H.: New stochastic approximation algorithms with adaptive step sizes, Optim. Lett. 6(8), 1831-1846 (2012)

[23] Xu, Z., Xu, X.: A new hybrid stochastic approximation algorithm, Optim. Lett. 7(3), 593-606 (2013)

[24] Yousefian, F., Nedic, A., Shanbhag, U. V.:On Stochastic Gradient and Subgradient Methods with Adaptive Steplength Sequences, Automatica J. IFAC 48(1), 56-67 (2012)

Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia.
E-mail address: milena.kresoja@dmi.uns.ac.rs

Department of Mathematics, Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University, Arhimedova 3, 1000 Skopje, Macedonia.
E-mail address: mdimovski16@gmail.com

Department of Mathematics, Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University, Arhimedova 3, 1000 Skopje, Macedonia.
E-mail address: irenatra@pmf.ukim.mk

Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia.
E-mail address: zorana@dmi.uns.ac.rs