Journal of Computational Mathematics
Vol.37, No.1, 2019, 76–94.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1710-m2017-0021

DESCENT DIRECTION STOCHASTIC APPROXIMATION ALGORITHM WITH ADAPTIVE STEP SIZES*

Zorana Lužanin
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Novi Sad, Serbia
Email: zorana@dmi.uns.ac.rs
Irena Stojkovska
Department of Mathematics, Faculty of Natural Sciences and Mathematics, Ss. Cyril and Methodius University, Skopje, Macedonia,
Email: irenatra@pmf.ukim.mk
Milena Kresoja
Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Novi Sad, Serbia
Email: milena.kresoja@dmi.uns.ac.rs

Abstract

A stochastic approximation (SA) algorithm with new adaptive step sizes for solving unconstrained minimization problems in noisy environment is proposed. New adaptive step size scheme uses ordered statistics of fixed number of previous noisy function values as a criterion for accepting good and rejecting bad steps. The scheme allows the algorithm to move in bigger steps and avoid steps proportional to $1/k$ when it is expected that larger steps will improve the performance. An algorithm with the new adaptive scheme is defined for a general descent direction. The almost sure convergence is established. The performance of new algorithm is tested on a set of standard test problems and compared with relevant algorithms. Numerical results support theoretical expectations and verify efficiency of the algorithm regardless of chosen search direction and noise level. Numerical results on problems arising in machine learning are also presented. Linear regression problem is considered using real data set. The results suggest that the proposed algorithm shows promise.

Mathematics subject classification: 90C15, 62L20.
Key words: Unconstrained optimization, Stochastic optimization, Stochastic approximation, Noisy function, Adaptive step size, Descent direction, Linear regression model.

1. Introduction

The main aim of the paper is to propose and analyse a new algorithm with adaptive step sizes for solving stochastic optimization problems. The problem under our consideration is an unconstrained minimization problem in noisy environment,

$$\min_{x\in\mathbb{R}^n} f(x), \tag{1.1}$$

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 observations of the objective function $f(x)$ and its gradient

* Received February 7, 2017 / Revised version received June 12, 2017 / Accepted October 23, 2017 / Published online August 14, 2018 /

$\nabla f(x) = g(x)$ are available for all $x \in \mathbb{R}^n$. Denote by ξ and ε random variable and random vector, respectively, defined on a probability space $(\Omega, \mathcal{F}, P)$. The noisy function and noisy gradient at each $x \in \mathbb{R}^n$, in this set-up, are given by

$$F(x) = f(x) + \xi \quad \text{and} \quad G(x) = g(x) + \varepsilon, \tag{1.2}$$

where ξ and ε represent the random noise terms. Also, we denote by $x^* \in \mathbb{R}^n$ a stationary point of $f(x)$ in (1.1), that is $g(x^*) = 0$. Throughout the paper we will use the following notation

$$\begin{aligned} F_k &= F(x_k) = f(x_k) + \xi_k = f_k + \xi_k \\ G_k &= G(x_k) = g(x_k) + \varepsilon_k = g_k + \varepsilon_k, \end{aligned} \tag{1.3}$$

where x_k is kth iteration. Index k used with ε and ξ allows us to consider the cases when the noise-generating process may change with k. We will refer the standard deviation of the noise term ε as *noise level*.

The most common method for solving problem (1.1) is *Stochastic Approximation* (SA) algorithm proposed by Robbins and Monro, [16]. It is introduced for finding roots of one-dimensional nonlinear scalar function and later extended to multidimensional systems by Blum, [2]. Iterative rule of SA algorithm is motivated by the gradient direction method and uses only noisy gradient observations. For a given initial iteration x_0, iterative rule is given by the formula

$$x_{k+1} = x_k - a_k G_k, \tag{1.4}$$

where $a_k > 0$ is a step size and G_k is the noisy gradient at x_k defined by (1.3). The sequence $\{a_k\}$ is called the *sequence of step size lengths* or *gain sequence*. The convergence of SA method is achievable in a stochastic sense under certain assumptions. Robbins and Monro established mean square (m.s.) convergence, [16], while almost sure (a.s.) convergence is established by Chen, [7] and Spall, [18]. They proved that method (1.4) converges to a solution of the system $g(x) = 0$.

The performance of SA method depends mostly on the choice of the step size sequence. Numerous modifications of SA algorithm based on the step size selection are proposed to improve the optimization process. Kesten, [9], proposed an accelerated SA algorithm, for one dimensional case, with the step sizes that depend on the frequency of sign changes of the differences between two successive iterations. The a.s. convergence of the accelerated SA algorithm is established. The method is extended for multidimensional problems and a.s. convergence is proved by Delyon and Juditsky, [8]. Idea of monitoring sign is further studied by Xu and Dai, [21]. An algorithm with adaptive step sizes is proposed by Yousefian et al., [22] where authors propose a scheme for minimizing strongly convex differentiable functions in noisy environment. The scheme generates a step size sequence that is a decreasing piecewise-constant function with a decrease that occurs when a suitable threshold error is met. SA algorithm with a line-search is proposed by Krejić et al., [10]. A line search along the negative gradient direction is applied while the iterates are far away from the solution and upon reaching some neighbourhood of the solution the method switches to SA rule. Approach in [10] is recently extended to general descent direction case by Krejić et al., [11]. This result allows application of faster, second-order methods while keeping the almost sure convergence. Algorithms that use second-order directions are frequently applied for solving large-scale problems in machine learning. SA algorithm with a quasi-Newton direction is successfully applied in [4–6]. A stochastic quasi-Newton method for solving nonconvex stochastic optimization problems is also proposed

in [19]. An adaptive step size algorithm with a general descent direction is recently proposed by Kresoja et al., [12]. The algorithm adjust steps sizes based on an interval around the mean of fixed number of previously observed noisy function values.

In this paper we propose a SA algorithm with a new adaptive step size scheme. Motivated by the scheme proposed in [12], we suggest a new criterion for the step size adoption which also uses only noisy function values. The new criterion is formed using a minimum and a maximum instead of mean of previous noisy function values and can be applied without knowing the true or estimated value of the noise level. The algorithm uses a general descent direction as search direction. Almost sure convergence is established, and numerical experiments are conducted.

The paper is organized as follows. Section 2 contains a brief overview of SA algorithms with gradient and descent direction separately, along with some of the existing stochastic approximation algorithms with adaptive step sizes. The detailed description and analysis of the new step size scheme, the corresponding algorithm, and the convergence analysis of the proposed algorithm are presented in Section 3. In Section 4, practical implementation issues are discussed and results from the numerical experiments are given. The method is tested using both, synthetic and real data. The conclusions are drawn in Section 5.

2. Preliminaries

2.1. Stochastic Approximation with Gradient Direction

In this subsection we will review the conditions for almost sure convergence of SA algorithm (1.4). The convergence conditions for the sequence $\{a_k\}$ are the following

$$a_k > 0,\ \sum_k a_k = \infty \text{ and } \sum_k a_k^2 < \infty. \tag{2.1}$$

The conditions (2.1) imply that the step size sequence should not decay neither too fast, nor too slow. One of the most used sequence is generalization of scaled harmonic sequence,

$$a_k = \frac{a}{(k+1+A)^\alpha}, \tag{2.2}$$

where $a > 0$, $A \geq 0$ and $0.5 < \alpha \leq 1$.

Denote by $\{x_k\}$ a sequence generated by SA method (1.4) and by $\mathcal{F}_k$ the σ-algebra generated by $x_0, x_1, \ldots, x_k$. The set of standard assumptions which ensures the convergence of SA algorithm is the following, [7].

A1 For any $\varepsilon > 0$ there exists $\beta_\varepsilon > 0$ such that

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

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 non-decreasing σ-algebras.

A3 There exists a constant $c > 0$ such that

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

Assumption A1 is the strong condition on the shape of $g(x)$, while the assumption A2 represents a classical zero mean condition in stochastic analysis. Under assumption A2, $G_k(x)$ is an unbiased estimator of the true gradient $g(x)$. Assumption A3 provides restrictions on the magnitude of $g(x)$, i.e. $\|g(x)\|^2$ and the second moment of observation noise cannot grow faster than a quadratic function of x.
Finally, we state the main convergence result for SA method.

Theorem 2.1. ([7]) *Assume that* A1-A3 *hold. Let* $\{x_k\}$ *be a sequence generated by SA method* (1.4)*, where the gain sequence* $\{a_k\}$ *satisfies the conditions* (2.1)*. Then the sequence* $\{x_k\}$ *converges to* x^* *a.s. for an arbitrary initial approximation* x_0*.*

2.2. Stochastic Approximation with Descent Direction

In this subsection we will review the convergence conditions for a descent direction form of SA algorithm proposed and analysed by Krejić et al., [11]. For a given initial approximation x_0, iterative rule of the algorithm is given by

$$x_{k+1} = x_k + a_k d_k, \tag{2.3}$$

where d_k is a descent direction defined by

$$G_k^T d_k < 0 \text{ a.s.}, \tag{2.4}$$

G_k is the noisy gradient at x_k and $\{a_k\}$ is a gain sequence that satisfies the conditions (2.1). The convergence of the descent direction method is also achievable in stochastic sense under a certain set of assumptions. Instead of assumption A1, two more assumptions on the direction d_k are imposed.

Let $\{x_k\}$ be a sequence generated by (2.3) and $\mathcal{F}_k$ the σ-algebra generated by $x_0, x_1, \ldots, x_k$. Additional assumptions needed for the convergence of SA algorithm with descent direction are the following [11].

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

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

A5 There exists $c_2 > 0$ such that

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

The assumption A4 limits the influence of the noise on d_k and it is analogous to the assumption C4 used in [17]. The assumption A5 connects the available noisy gradient with the descent direction. Taking $d_k = -G_k$, we get that A5 is satisfied for any $c_2 \geq 1$.

Theorem 2.2. ([11]) *Assume that* A2-A5 *hold. Let* $\{x_k\}$ *be a sequence generated by* (2.3)*. Then the sequence* $\{x_k\}$ *converges to* x^* *a.s. for an arbitrary initial approximation* x_0*.*

A descent direction form of SA method, is studied also by Bertsecas and Tsitsiklis in [1].

2.3. Stochastic Approximation with Adaptive Step Sizes

The main drawback of SA algorithms (1.4) and (2.3) is slow convergence which mostly depends on the choice of the step size sequence $\{a_k\}$. The step sizes proportional to $1/k$, such as steps (2.2), become small very fast and make the iterative process quite slow. In order to overcome this difficulty a number of modifications based on adaptive step size selection is proposed in the literature.

One of the first adaptive step size techniques is proposed by Kesten, [9]. It is based on the frequency of sign changes of the differences $x_{k+1} - x_k$. Frequent sign changes indicate that the current iteration is near the solution and a smaller step size is used in the next iterate. A larger step size is used if changes in the sign are not frequent. The following step size rule is proposed

$$a_k = \frac{a}{z_k + 1}, \tag{2.5}$$

where $a > 0$ and $z_{k+1} = z_k + \mathcal{I}(G_{k+1}^T G_k)$ and $\mathcal{I}$ represent indicator function defined by $\mathcal{I}(t) = 1$ if $t < 0$ and $\mathcal{I}(t) = 0$ if $t \geq 0$.

Kesten's idea is modified by Xu and Dai [21]. Authors discuss the properties of $\frac{z_k}{k}$ and propose a switching algorithm with the following step size rule

$$a_k = \begin{cases} \frac{a}{(k+1+A)^\alpha}, & \text{if } l_k \geq v, \\ \frac{a}{(k+1+A)^\beta}, & \text{if } l_k < v, \end{cases} \tag{2.6}$$

where $a > 0$, $A \geq 0$, $l_k = |\frac{z_k}{k} - P(\varepsilon_1^T \varepsilon_2 < 0)|$, $0.5 < \alpha < \beta \leq 1$, v is a small positive constant, and $\varepsilon_1, \varepsilon_2$ are the gradient noises defined by (1.3).

SA algorithm with adaptive step sizes and a general descent direction d_k defined by (2.4) is recently proposed in [12]. The step sizes are adjusted by analysing intervals for the optimal function value $f(x^*)$ at each iteration. Intervals are formed using fixed number of previously observed noisy function values. Tracking the observed values of the objective function may considerably improve the knowledge about the optimization process, even it might be more costly. This issue is also discussed in [17,20], where it is concluded that using observed function values to accept or reject steps can improve the algorithm's stability. The step size sequence is formed according to the rule

$$a_k = \begin{cases} a\theta^{s_k}, & F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma}, \\ 0, & F_k > \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma}, \\ \frac{a}{(t_k+1+A)^\alpha}, & \text{otherwise}, \end{cases} \tag{2.7}$$

where $m(k) = \min\{k, m\}$, $m \in \mathbb{N}$, $\theta \in (0,1)$, $a, \hat{\sigma} > 0$, $A \geq 0$, $0.5 < \alpha \leq 1$, s_k is a counter of the occurrences of the events,

$$\left\{ F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma} \right\},$$

and t_k is a counter of the occurrences of the events

$$\left\{ \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma} \leq F_k \leq \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma} \right\}.$$

Under additional assumption on the noise terms ξ_k, that is,

$$\begin{aligned}&\xi_k, k = 0, 1, 2, ... \text{ are i.i.d. continuous random variables with a common}\\&\text{probability density function (pdf) } p(x) > 0 \text{ a.s. for all } x \in \mathbb{R},\end{aligned} \tag{2.8}$$

almost sure convergence of SA algorithm with step sizes (2.7) is proven [12].

Theorem 2.3. ([12]) *Assume that* A2-A5 *hold, and the noise terms ξ_k satisfy the condition* (2.8). *Let $\{x_k\}$ be a sequence generated by* (2.3) *with the step sizes $\{a_k\}$ defined by* (2.7). *Then the sequence $\{x_k\}$ converges to x^* a.s. for an arbitrary initial approximation x_0.*

There is a justification that the constant $\hat{\sigma}$ in the step size rule (2.7) can be replaced with true or estimated standard deviation of the noise added to the functional values $F(x)$. Numerical experiments also showed that this is a quite right decision, [12].

3. New Stochastic Approximation Algorithm

3.1. The Step Size Selection Rule and the Algorithm

Motivated by (2.7), we propose a new adaptive step size rule for SA algorithm. Our main aim is to propose a criterion for accepting and rejecting steps with an approach that has a direct insight into whether the objective function is improving. We suggest using the minimum and the maximum of $m(k)$ previously observed noisy function values $F_{k-1}, \ldots, F_{k-m(k)}$ instead of their shifted mean. Throughout the paper we use the following notation:

$$F^{min}_{k,m(k)} = \min_{1\le j\le m(k)} F_{k-j} \text{ and } F^{max}_{k,m(k)} = \max_{1\le j\le m(k)} F_{k-j},$$

where $m(k) = \min\{k, m\}$ and $m \in \mathbb{N}$.

The formal formulation of our adaptive step size rule is the following

$$a_k = \begin{cases} a\theta^{s_k}, & F_k < F^{min}_{k,m(k)},\\ 0, & F_k > F^{max}_{k,m(k)},\\ \frac{a}{(t_k+1+A)^\alpha}, & \text{otherwise,}\end{cases} \tag{3.1}$$

where

- $\theta \in (0, 1)$, $a > 0$, $A \ge 0$, $0.5 < \alpha \le 1$,
- s_k counts occurrences of the events $\left\{F_k < F^{min}_{k,m(k)}\right\}$ up to kth iteration,
- t_k counts occurrences of the events $\left\{F^{min}_{k,m(k)} \le F_k \le F^{max}_{k,m(k)}\right\}$ up to kth iteration.

Using the rule (3.1), if the observed (noisy) function value in kth iteration F_k, defined by (1.3), is higher than the maximum of $m(k)$ previously observed function values, we suggest blocking the step by taking $a_k = 0$. If F_k is lower than the minimum of $m(k)$ previously observed function values, we suggest step size $a_k = a\theta^{s_k}$ in the next iteration. Otherwise, if F_k is between minimum and maximum of $m(k)$ previously observed function values, we propose backup step size similar to the step size (2.2), substituting k with t_k which counts the occurrences of the mentioned events.

Our initial idea was to use a constant full step size $a_k = 1$ when there is an improvement in the function value. However, we chose the sequence $a_k = a\theta^{s_k}$ which retains property of the gain sequence $\{a_k\}$, suitable for convergence analysis. This step size sequence of larger steps showed good numerical results in [12], which encouraged us to keep it in the new step size rule. As it will be demonstrated in Section 4, we recommend taking θ close to 1. Note that the parameter θ is the key parameter in controlling the magnitude of the step size when good scenario occurs. The step size $a_k = a\theta^{s_k}$ with $\theta \approx 1$ will produce longer steps than steps of the SA form (2.2), while the iterates are far away from the solution, but also when the number of iterates becomes large.

Recall that the scheme (2.7) estimates the optimal function value in each iterate by forming an interval using $m(k)$ previous noisy function values

$$J_k = (\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma}, \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma}), \tag{3.2}$$

where $\hat{\sigma} > 0$ is a constant, and $m(k) = \min\{k, m\}$, $m \in \mathbb{N}$. An optimal $\hat{\sigma}$ in (3.2) is related to the noise level of the function measurements which is unknown in practice and has to be estimated, sometimes by additional procedures. The scheme (3.1) constructs an interval of the following form

$$\tilde{J}_k = (F^{min}_{k,m(k)}, F^{max}_{k,m(k)}) \tag{3.3}$$

at each iterate. This approach can be applied without knowing or estimating the noise because it does not require parameter $\hat{\sigma}$ in the (3.3). The both intervals J_k and $\tilde{J}_k$, estimate the optimal function value independently. They are both sensitive on the extreme noisy function values, but correct them in a different manner. The interval J_k has more variable bounds, the extreme function values influence the mean, but the constant $\hat{\sigma}$ corrects the influence. On the other hand, the extreme function values have the biggest influence on the bound of the interval $\tilde{J}_k$, and can induce periods of a constant bound during the optimization process, which helps to capture the optimal value when approaching to the solution.

Finally, we give the formulation of algorithm based on the adaptive step size selection rule (3.1).

Algorithm 3.1. Min-Max Adaptive Stochastic Approximation

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

Step 1. Direction selection. Choose d_k such that (2.4) holds.

Step 2. Step size selection. Calculate the noisy function measurement F_k and select the step size a_k according to the rule (3.1).

Step 3. Update iteration. Calculate $x_{k+1} = x_k + a_k d_k$, set $k = k + 1$ and go to Step 1.

3.2. Convergence Analysis

We will show that the step size sequence $\{a_k\}$ generated by Algorithm 3.1 satisfies the conditions (2.1) almost surely. We assume that noise terms ξ_k, $k = 0, 1, 2, \ldots$ satisfy the

conditions (2.8).

Firstly, we will focus on the distribution of the step sizes. It depends on the probability of the events $\left\{F_k > F^{max}_{k,m(k)}\right\}$, $\left\{F_k < F^{min}_{k,m(k)}\right\}$ and $\left\{F^{min}_{k,m(k)} \leq F_k \leq F^{max}_{k,m(k)}\right\}$. Recall that for each k, $F_k = f_k + \xi_k$, where $f_k = f(x_k)$ is the true function value at x_k. The following lemma holds.

Lemma 3.1. *If the noise terms ξ_k are i.i.d. continuous random variables and $f_k = f_{k-j},\ j = 1, \ldots, m(k)$, then the following probabilities hold*

$$P(F_k > F^{max}_{k,m(k)}) = \frac{1}{m(k)+1}, \tag{3.4}$$

$$P(F_k < F^{min}_{k,m(k)}) = \frac{1}{m(k)+1}, \tag{3.5}$$

$$P(F^{min}_{k,m(k)} \leq F_k \leq F^{max}_{k,m(k)}) = \frac{m(k)-1}{m(k)+1}. \tag{3.6}$$

Proof. Let us denote by $\Phi(x)$ the cumulative distribution function (cdf) of any of the random variables ξ_k. If we denote by $\Phi^k_j(x)$ the cdf of the random variable F_{k-j}, then from $F_{k-j} = f_{k-j} + \xi_{k-j}$, we have that

$$\begin{aligned}\Phi^k_j(x) =& P(F_{k-j} \leq x) = P(f_{k-j} + \xi_{k-j} \leq x)\\ =& P(\xi_{k-j} \leq x - f_{k-j}) = \Phi(x - f_{k-j}).\end{aligned} \tag{3.7}$$

And, if we denote by $\Phi^k_{(m(k))}(x)$ the cdf of the random variable $F^{max}_{k,m(k)}$, then from the iid property of the noise terms we have that F_{k-j}, $j = 1, \ldots, m(k)$ are also independent continuous random variables, so this, the equality (3.7) and the assumption $f_k = f_{k-j},\ j = 1, \ldots, m(k)$ imply that

$$\begin{aligned}\Phi^k_{(m(k))}(x) =& P(F^{max}_{k,m(k)} \leq x) = P(F_{k-1} \leq x, \cdots, F_{k-m(k)} \leq x)\\ =& P(F_{k-1} \leq x) \cdots P(F_{k-m(k)} \leq x) = \Phi^k_1(x) \cdots \Phi^k_{m(k)}(x)\\ =& \Phi(x - f_{k-1}) \cdots \Phi(x - f_{k-m(k)}) = (\Phi(x - f_k))^{m(k)}.\end{aligned} \tag{3.8}$$

We will use that for any two independent continuous random variables X and Y with cdfs $\Phi_X(x)$ and $\Phi_Y(x)$ respectively, the probability of the event $\{X > Y\}$ can be expressed as

$$P(X > Y) = \int_{-\infty}^{+\infty} \Phi_Y(x)\Phi'_X(x)dx. \tag{3.9}$$

So, (3.7)-(3.9) and the independence of the random variables F_k and $F^{max}_{k,m(k)}$ imply that

$$P(F_k > F^{max}_{k,m(k)}) = \int_{-\infty}^{+\infty} (\Phi(x - f_k))^{m(k)}\Phi'(x - f_k)dx = \int_0^1 y^{m(k)}dy = \frac{1}{m(k)+1},$$

since $\Phi(x)$ is a cdf and $\lim_{x\to-\infty} \Phi(x) = 0$ and $\lim_{x\to+\infty} \Phi(x) = 1$. Similarly, it can be derived that

$$P(F_k < F^{min}_{m(k)})) = \frac{1}{m(k)+1}.$$

And finally,

$$P(F^{min}_{k,m(k)} \leq F_k \leq F^{max}_{k,m(k)}) = 1 - \frac{2}{m(k)+1} = \frac{m(k)-1}{m(k)+1},$$

which completes the proof. □

Remark 3.1. Note that if the noise terms ξ_k are i.i.d. continuous random variables and there are $m(k)$ consecutive zero steps $a_{k-1} = a_{k-2} = \ldots = a_{k-m(k)} = 0$, then $x_k = x_{k-1} = \ldots = x_{k-m(k)}$ so $f_k = f_{k-j}$ for $j = 1, \ldots, m(k)$. Therefore, Lemma 3.1 holds.

Remark 3.1 helps us to recognize the importance of the event

$$A_k = \left\{a_{k-1} = a_{k-2} = \ldots = a_{k-m(k)} = 0\right\} \tag{3.10}$$

for the distribution of the step sizes a_k. So, our next step will be to investigate the probability of having $m(k)$ consecutive zero steps.

Lemma 3.2. *Let the step sizes a_k be defined by* (3.1)*. If the noise terms ξ_k satisfy the conditions* (2.8)*, then for $k = 1, 2, \ldots$, the following inequality holds*

$$P(A_k) > 0, \tag{3.11}$$

where A_k is defined by (3.10)*.*

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

$$P(A_k) = 0. \tag{3.12}$$

It follows

$$\begin{aligned} 0 = P(A_k) &= P(F_{k-1} > F^{max}_{k-1,m(k)}, \ldots, F_{k-m(k)} > F^{max}_{k-m(k),m(k)}) \\ &= P(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > F^{max}_{k-m(k),m(k)}) \\ &\geq P\left(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > \ldots > F_{k-2m(k)}\right). \end{aligned} \tag{3.13}$$

Therefore, we have

$$P\left(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > \ldots > F_{k-2m(k)}\right) = 0. \tag{3.14}$$

Let us now define *δ-neighbourhood of the optimal value* $f^* = f(x^*)$. We say, y is in δ-neighbourhood of the optimal value f^* if $|y - f^*| < \delta$, where $\delta > 0$. Next, denote by $B_{\frac{\delta}{2}}$ the event

$$B_{\frac{\delta}{2}} = \left\{ f_{k-j} \text{ is in } \frac{\delta}{2} - \text{neighbourhood of the optimal value } f^*,\ j = 1, \ldots, 2m(k) \right\}.$$

The event $B_{\frac{\delta}{2}}$ represents the situation when $2m(k)$ consecutive true values of the objective function are in some $\frac{\delta}{2}$-neighbourhood of the optimal value f^*. Note that the event $B_{\frac{\delta}{2}}$ depends on the index k, although we omit the k in the notation. The reason is that at the beginning of the proof we assume the existence of k such that $P(A_k) = 0$, therefore for the remaining proof, k acts as a constant.
Now, we chose $\delta > 0$ such that

$$P(B_{\frac{\delta}{2}}) > 0. \tag{3.15}$$

Note that, such $\delta > 0$ exists. For example, we can take

$$\delta = 2 * \max_{1 \leq j \leq 2m(k)} |f_{k-j} - f^*| + 1.$$

For this choice of δ, actually we have $P(B_{\frac{\delta}{2}}) = 1$.

It follows that

$$\begin{aligned} 0 &= P\left(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > \ldots > F_{k-2m(k)}\right) \\ &\geq P\left(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > \ldots > F_{k-2m(k)} | B_{\frac{\delta}{2}}\right) P(B_{\frac{\delta}{2}}). \end{aligned} \tag{3.16}$$

So, from (3.15) and (3.16) we obtain

$$P\left(F_{k-1} > F_{k-2} > \ldots > F_{k-m(k)} > \ldots > F_{k-2m(k)} | B_{\frac{\delta}{2}}\right) = 0. \tag{3.17}$$

However, if f_{k-j}, $j = 1, 2, \ldots, 2m(k)$ are in a $\frac{\delta}{2}-$neighbourhood of the optimal value f^*, then we have

$$|f_{k-j} - f_{k-i}| \leq |f_{k-j} - f^*| + |f^* - f_{k-i}| < \frac{\delta}{2} + \frac{\delta}{2} = \delta,$$

for all $j, i = 1, 2, \ldots, 2m(k)$ and

$$f_{k-i} - \delta < f_{k-j} < f_{k-i} + \delta.$$

Under the realization of the event $B_{\frac{\delta}{2}}$, the inequalities

$$\xi_{k-j} > \xi_{k-j-1} + \delta,\ j = 1, 2, \ldots, 2m(k) - 1 \tag{3.18}$$

imply

$$\begin{aligned} F_{k-j} &= f_{k-j} + \xi_{k-j} > f_{k-j} + \xi_{k-j-1} + \delta \\ &> f_{k-j-1} + \xi_{k-j-1} = F_{k-j-1},\ j = 1, 2, \ldots, 2m(k) - 1. \end{aligned} \tag{3.19}$$

Consequently,

$$\begin{aligned} &P\left(F_{k-j} > F_{k-j-1},\ j = 1, 2, \ldots, 2m(k) - 1 | B_{\frac{\delta}{2}}\right) \geq \\ &P\left(\xi_{k-j} > \xi_{k-j-1} + \delta,\ j = 1, 2, \ldots, 2m(k) - 1 | B_{\frac{\delta}{2}}\right). \end{aligned} \tag{3.20}$$

Now, (3.17) and (3.20) imply that

$$P\left(\xi_{k-j} > \xi_{k-j-1} + \delta,\ j = 1, 2, \ldots, 2m(k) - 1 | B_{\frac{\delta}{2}}\right) = 0. \tag{3.21}$$

Taking into account that the conditional probability in (3.21) is independent of the condition, we can rewrite (3.21) as

$$P\left(\xi_{k-j} > \xi_{k-j-1} + \delta,\ j = 1, 2, \ldots, 2m(k) - 1\right) = 0. \tag{3.22}$$

Note that

$$\begin{aligned} I(\delta) &= P\left(\xi_{k-j} > \xi_{k-j-1} + \delta,\ j = 1, 2, \ldots, 2m(k) - 1\right) \\ &= P\left(\xi_{k-1} > \xi_{k-2} + \delta > \xi_{k-3} + 2\delta > ... > \xi_{k-2m(k)} + (2m(k) - 1)\delta\right) \\ &= \int_{-\infty}^{\infty} p(x_{k-1}) \int_{-\infty}^{x_{k-1}-\delta} p(x_{k-2}) \cdots \\ &\quad \int_{-\infty}^{x_{k-2m(k)+1}-(2m(k)-1)\delta} p(x_{k-2m(k)}) dx_{k-1} dx_{k-2} \ldots dx_{k-2m(k)} > 0 \end{aligned} \tag{3.23}$$

almost surely for all $\delta > 0$, since $p(x) > 0$ a.s. by conditions (2.8), and $I(\delta)$ is a decreasing function, with

$$\lim_{\delta \to 0} I(\delta) = \frac{1}{(2m(k))!} \quad \text{and} \quad \lim_{\delta \to +\infty} I(\delta) = 0,$$

which is in contradiction with (3.22). This implies that $P(A_k) > 0$ for all k. □

Now, when we know that $m(k)$ consecutive zero steps may occur with non zero probability, we can show that there is non zero probability of occurring each of the steps $a_k = a\theta_k^s$, $a_k = 0$ and $a_k = \frac{a}{(t_k+1+A)^\alpha}$ at every iteration k.

Lemma 3.3. *Let the step sizes a_k be defined by* (3.1)*. If the noise terms ξ_k satisfy the condition* (2.8)*, then for all $k = 1, 2, \ldots$*

$$P(a_k = a\theta^{s_k}) > 0,\ P(a_k = 0) > 0 \ \text{ and } \ P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}) > 0. \tag{3.24}$$

Proof. From Remark 3.1 and Lemma 3.1, it follows

$$P(a_k = a\theta^{s_k}) \geq P(a_k = a\theta^{s_k}|A_k) \cdot P(A_k) = \frac{1}{m(k)+1} \cdot P(A_k) > 0, \tag{3.25}$$

$$P(a_k = 0) \geq P(a_k = 0|A_k) \cdot P(A_k) = \frac{1}{m(k)+1} \cdot P(A_k) > 0, \tag{3.26}$$

$$\begin{aligned} P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}) &\geq P(a_k = \frac{a}{(t_k + 1 + A)^\alpha}|A_k) \cdot P(A_k) \\ &= \frac{m(k)-1}{m(k)+1} \cdot P(A_k) > 0. \end{aligned} \tag{3.27}$$

Note that the conditional probabilities $P(\cdot|A_k)$ are well defined because of Lemma 3.2. □

Lemma 3.3 ensures that there are infinitely many of both of non zero steps almost surely.

Lemma 3.4. *Let the step sizes a_k be defined by* (3.1) *and let the noise terms ξ_k satisfy the conditions* (2.8)*. Then there are almost surely infinitely many steps $a_k = a\theta^{s_k}$ and infinitely many steps $a_k = \frac{a}{(t_k+1+A)^\alpha}$.*

Proof. Same as the proof of Lemma 3.3 in [12]. □

Lemma 3.4 ensures that the step size sequence $\{a_k\}$ satisfies almost surely the conditions (2.1).

Theorem 3.1. *If the noise terms ξ_k satisfy the conditions* (2.8)*, then the step size sequence $\{a_k\}$, defined by* (3.1)*, satisfies the conditions* (2.1) *almost surely.*

Proof. If we denote by $C = \{k|F_k < F^{min}_{k,m(k)}\}$ and $D = \{k|F^{min}_{k,m(k)} \leq F_k \leq F^{max}_{k,m(k)}\}$, then by the definition of the sequence $\{a_k\}$, the step size selection rule (3.1), we have

$$\begin{aligned} \sum_k a_k &= \sum_{k \in C} a\theta^{s_k} + \sum_{k \in D} \frac{a}{(t_k + 1 + A)^\alpha} = \sum_k a\theta^k + \sum_k \frac{a}{(k + 1 + A)^\alpha} = \infty, \\ \sum_k a_k^2 &= \sum_{k \in C} (a\theta^{s_k})^2 + \sum_{k \in D} (\frac{a}{(t_k + 1 + A)^\alpha})^2 = \sum_k (a\theta^k)^2 + \sum_k (\frac{a}{(k + 1 + A)^\alpha})^2 < \infty, \end{aligned}$$

almost surely, since almost surely we have infinitely many steps $a_k = a\theta^{s_k}$ and infinitely many steps $a_k = \frac{a}{(t_k+1+A)^\alpha}$ by Lemma 3.4. So, the step size sequence $\{a_k\}$ satisfies the conditions (2.1) almost surely. □

Now, we will establish the convergence of the Algorithm 3.1. Moreover, we will discuss cases when descent direction is a negative gradient and a general descent direction separately.

Note that conditions (2.1) for the step sizes in SA convergence theorems, Theorem 2.1 and Theorem 2.2, are stated for deterministic step sizes a_k. When step sizes are random, the conditions (2.1) need to be satisfied almost surely (a.s.). Moreover, it is necessary 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. This means that we are not allowed to use information from $(k+1)$th iteration to compute a_k, similar to the assumption in [14]. Under these additional assumptions, the SA convergence theorems, Theorem 2.1 and Theorem 2.2, also hold when step sizes a_k are random.

Theorem 3.1 and Theorem 2.2 ensure the almost sure convergence of Algorithm 3.1 with a general descent direction.

Theorem 3.2. *Assume that* A2-A5 *hold. Let* $\{x_k\}$ *be a sequence generated by Algorithm* 3.1*, where the noise terms* ξ_k *satisfy the conditions* (2.8)*. Then the sequence* $\{x_k\}$ *converges to* x^* *a.s. for an arbitrary initial approximation* x_0*.*

The almost sure convergence of the Algorithm 3.1 when $d_k = -G_k$ can be established using SA convergence theorem, Theorem 2.1, and the property of the gain sequence $\{a_k\}$ given with Theorem 3.1.

Corollary 3.1. *Assume that* A1-A3 *hold. Let* $\{x_k\}$ *be a sequence generated by Algorithm* 3.1 *with* $d_k = -G_k$*, where the noise terms* ξ_k *satisfy the conditions* (2.8)*. Then the sequence* $\{x_k\}$ *converges to* x^* *a.s. for an arbitrary initial approximation* x_0*.*

4. Numerical Experiments

4.1. Testing the algorithms on synthetic data

Algorithm 3.1 is tested using different search directions and compared with other relevant algorithms. The collection of test problems consists of 20 problems. Detailed list of the test functions, the problem dimensions n, the initial approximations x_0 and optimal function value f^* is given in Table 1. The problems are selected from the collections of unconstrained minimization problems in [3], which are also mainly described in [13] and [15]. First 18 test problems are given in the form of nonlinear least squares,

$$f(x) = \sum_{i=1}^{r} f_i^2(x).$$

The transformation of original problems into problems in noisy environment is performed by adding normal distributed noise to the function and gradient evaluations, i.e. the noise of the form

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

where σ is the noise level and $I_{n\times n}$ is the identity matrix. The objective function and the gradient value at the current iterate x_k are calculated using sample average approximation of

Table 4.1: Test problems.

No	Problem; n	x_0	f^*
1	The Gaussian function; 3	$(4/10, 1, 0)$	1.12793×10^{-8}
2	The Box 3-dimensional function; 3	$(0, 10, 5)$	0
3	The variably dimensioned function; 4	$(3/4, 2/4, 1/4, 0)$	0
4	The Watson function; 4	$(0, 0, 0, 0)$	2.4384×10^{-6}
5	The Penalty Function I; 10	$(1, 1, \ldots, 1)$	7.08765×10^{-5}
6	The Penalty Function II; 4	$(1/2, 1/2, 1/2, 1/2)$	9.37629×10^{-6}
7	The Trigonometric Function; 10	$(1/10, 1/10, \ldots, 1/10)$	0
8	The Beale Function; 2	$(1, 1)$	0
9	The Chebyquad Function; 10	$(5/11, 10/11 \ldots, 50/11)$	6.50395×10^{-3}
10	The Gregory and Karney Function; 4	$(0, 0, 0, 0)$	-4
11	The Hilbert Matrix Function; 4	$(1, 1, 1, 1)$	0
12	The De Jong Function 1; 3	$(-5.12, 0, 5.12)$	0
13	The Branin RCOS Function; 2	$(-1, 1)$	0.397887
14	The Colville Polynomial; 4	$(1/2, 1, -1/2, -1)$	0
15	The Powell 3D Function; 3	$(0, 1, 2)$	1
16	The Himmelblau function, 2	$(-1.3, 2.7)$	0
17	The Fletcher-Powell function; 3	$(0, 0, 0)$	0
18	The Biggs EXP6 function; 6	$(1, 2, 1, 1, 1, 1)$	0
19	Strictly Convex 1; 10	$(1/10, 2/10, \ldots, 1)$	10
20	Strictly Convex 2; 10	$(1, 1, \ldots, 1)$	5.5

a small size equals to 3. Testing procedure is motivated by computational implementation in [11]. For each test problem and each algorithm, $N = 50$ independent runs starting from the same initial point are conducted. Exit parameters are the final iteration x_{end}, the final function value F_{end}, and the final gradient value G_{end}. Algorithms stop if the gradient value becomes small enough, $||G_k|| \leq C$, where we use $C = \min\{\sqrt{n}\sigma, 1\}$, or if the maximal number of $200n$ function evaluations is reached, with each gradient evaluation counted as n function evaluations. In this manner, the algorithms stop with x_{end} if either we reach a stationary point in stochastic sense or if the maximal number of function evaluations is used. Runs are classified into three categories: successful (convergent), partially successful and unsuccessful (divergent) runs. A run is successful if a method stops due to $||G_{end}|| \leq C$. The number of successful runs is denoted by $Nconv$. If $||G_{end}|| > 200\sqrt{n}$, the run is unsuccessful. The number of divergent runs is denoted by $Ndiv$. A run that stops due to exhausting the maximal number of allowed function evaluations is partially successful and the number of these runs is denoted by $Npar$. Algorithm 3.1 is tested with a negative gradient direction and with a quasi-Newton direction. We have chosen BFGS direction $d_k = -B_k^{-1}G_k$, with the update formula

$$B_{k+1} = B_k - \frac{B_k \delta_k \delta_k^T B_k}{\delta_k^T B_k \delta_k} + \frac{\Delta_k \Delta_k^T}{\Delta_k \delta_k}, \tag{4.1}$$

where

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

Note that the gradient difference Δ_k is calculated using the same sample set which is also implemented in [6, 11, 12, 19].

Since we consider "zero" step as a bad scenario, during testing procedure we have put limitation to the number of consecutive zero steps. In theoretical analysis, we have shown that step size sequence has three infinite subsequences almost surely, see Lemma 3.4. It means there cannot occur infinitely many consecutive zero steps. Therefore, the limitation of the number of consecutive zero steps has a theoretical justification. A correction is done using following rule: if the number of consecutive zero steps is greater than some predetermined number m_{corr}, in next iteration we use $a_k = \frac{a}{(t_k+1+A)^\alpha}$ as a step size. We have obtained empirically that it is the best to use $m_{corr} = m + 1$ as a correction value.

The values of parameters a, A and α that we use in all tested algorithms are given in Table 4.2.

Table 4.2: 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
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	1	0	0.602
18	1	0	0.602
19	0.5	100	0.501
20	0.1	100	0.75

4.1.1. Sensitivity analysis

We analyze sensitivity of the Algorithm 3.1 with $d_k = -G_k$ (MMGD) and with $d_k = -B_k^{-1}G_k$ (MMDD) with respect to parameter θ for different levels of noise. Two values are chosen for $\theta = 0.75, 0.999$. Similarly as in [12], we obtained empirically $m = 10$ as the most suitable choice and used this value in testing procedures.

As a tool for the sensitivity analysis we use Mean Squared Error (MSE) of the objective function estimator given by

$$MSE(f) = \sum_{r:||G^r||\leq C} (y^r - f^*)^2 / N_{conv},$$

where G^r is the last estimate of the gradient, y^r is the last estimate of the optimal functional value f^*.

Table 4.3 shows Mean Squared Error obtained by performing algorithms MMGD and MMDD with $\theta = 0.75$ and $\theta = 0.999$ for problems 1-10 tested with noise levels $\sigma = 0.4$ and $\sigma = 1$. The results for problems 11-20 are listed in Table 4.4. Note that *fail* denotes a case when all runs either partially successful or divergent.

Table 4.3: MSE(f) for Problems 1-10.

prb	σ	MMGD		MMDD	
		$\theta = 0.75$	$\theta = 0.999$	$\theta = 0.75$	$\theta = 0.999$
1	0.4	7.20E-05	5.00E-05	2.42E-04	2.47E-04
	1	1.46E-03	2.59E-03	8.16E-04	2.47E-04
2	0.4	3.38E-04	1.80E-05	1.79E+01	8.33E-06
	1	9.80E-05	1.28E-04	1.30E-01	7.37E-04
3	0.4	fail	fail	fail	5.94E-04
	1	fail	fail	fail	2.88E-02
4	0.4	1.69E-03	7.75E-04	fail	fail
	1	2.46E-03	3.31E-03	fail	fail
5	0.4	fail	fail	fail	fail
	1	fail	fail	fail	fail
6	0.4	2.90E-04	1.80E-05	6.88E-03	3.11E-03
	1	8.03E-04	2.88E-04	2.70E-02	2.23E-01
7	0.4	3.20E-05	1.28E-04	2.32E-04	1.28E-04
	1	fail	fail	fail	fail
8	0.4	2.63E-04	fail	7.78E+00	9.62E-01
	1	2.59E-03	fail	2.94E+01	7.22E-01
9	0.4	3.78E-05	3.10E-05	3.26E-03	5.83E-04
	1	fail	fail	6.97E-03	6.97E-03
10	0.4	2.45E-01	1.29E-01	6.57E-01	5.32E-01
	1	2.44E-01	1.28E-01	3.57E-01	3.75E+00

According to the obtained results, the performance of the Algorithm 3.1 is sensitive to the parameter θ. Taking larger θ decreases $MSE(f)$ in almost all cases for smaller level of noise, regardless of chosen direction. This result confirms our initial hypothesis for taking larger step when a sufficient decrease of objective function value is observed. When the noise level is higher, $\sigma = 1$, taking larger θ does not produce such clear pattern in reduction of $MSE(f)$. Therefore, when noise have strong influence, it may be useful to take smaller θ in some cases. It will still produce larger steps when good scenario occurs at the beginning of the process.

4.1.2. Comparison of the algorithms

Now, we compare performance of the Algorithm 3.1 with algorithms presented in Section 2. Comparative results for the following 7 algorithms are presented:

- SAGD - Algorithm (1.4) with SA step sizes (2.2), [16]
- XDGD - Algorithm (1.4) with adaptive step sizes (2.6), [21]
- MSGD - Algorithm (1.4) with adaptive step size rule (2.7), negative gradient direction $d_k = -G_k$, $\theta = 0.999$, $m = 10$ and and $\hat{\sigma} = \sigma$, [12]
- MMGD - Algorithm 3.1 with negative gradient direction $d_k = -G_k$, $\theta = 0.999$ and $m = 10$

Table 4.4: MSE(f) for Problems 11-20.

prb	σ	MMGD		MMDD	
		$\theta = 0.75$	$\theta = 0.999$	$\theta = 0.75$	$\theta = 0.999$
11	0.4	5.71E-03	5.78E-04	2.24E-01	1.05E-01
	1	1.04E-02	1.06E-03	3.69E+00	8.88E-02
12	0.4	fail	4.50E-04	3.87E+01	9.88E-01
	1	fail	1.57E-03	2.01E+02	1.52E+01
13	0.4	9.78E-06	1.24E-06	1.93E-03	3.45E-01
	1	4.59E-05	6.10E-03	1.88E+01	9.34E-01
14	0.4	fail	fail	fail	fail
	1	fail	fail	fail	fail
15	0.4	3.06E-04	2.00E-03	1.48E-02	1.61E-02
	1	1.25E-03	1.11E-02	2.30E-02	1.67E-03
16	0.4	6.91E-03	fail	fail	1.44E-02
	1	5.53E-03	fail	fail	fail
17	0.4	fail	fail	1.98E-01	4.56E+00
	1	fail	fail	8.74E+01	8.44E-01
18	0.4	fail	2.65E-03	2.08E+01	3.24E-03
	1	2.59E-03	5.02E-03	5.69E-02	1.16E-01
19	0.4	3.38E-04	5.12E-04	3.44E-03	2.00E-04
	1	3.20E-03	7.84E-02	fail	1.27E+01
20	0.4	3.43E-01	8.82E-04	5.15E-02	2.37E-03
	1	fail	2.12E-01	2.62E+01	fail

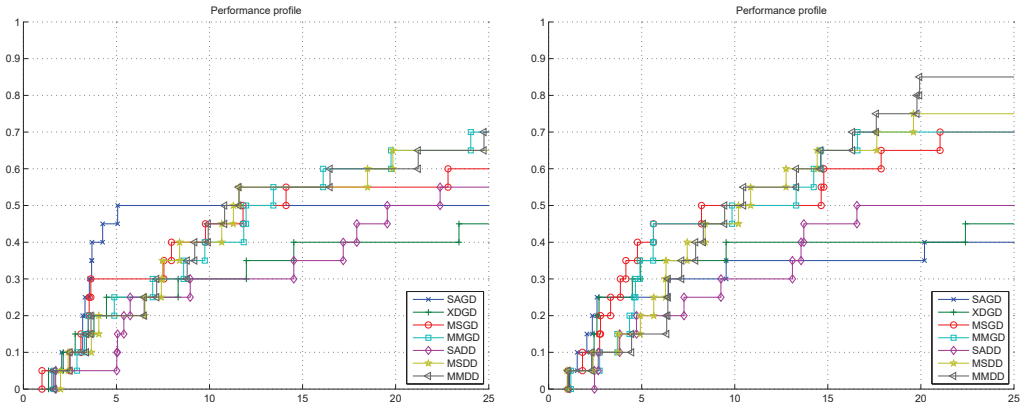
- SADD - Algorithm (2.3) with BFGS direction and SA step sizes (2.2), [11]
- MSDD - Algorithm (2.3) with adaptive step size rule (2.7), BFGS direction, $\theta = 0.999$, $m = 10$ and $\hat{\sigma} = \sigma$, [12]
- MMDD - Algorithm 3.1 with BFGS direction, $\theta = 0.999$ and $m = 10$

We have chosen to use $m = 10$ and $\theta = 0.999$ for the both step size rules, (3.1) and (2.7) compare these algorithms. For the performance measure we use the number of function evaluation needed in successful and partially successful runs

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

where $Ncon_{ij}$ is the number of successful runs for ith algorithm to solve problem j, $Npar_{ij}$ is the number 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, $i = 1, \cdots, 7,\ j = 1, \cdots, 20,\ r = 1, \cdots, 50$.

Figure 4.1 shows performance profiles for $\sigma = 0.4$ and $\sigma = 1$. For both noise levels, Algorithm 3.1 outperforms all other tested algorithms, regardless of chosen direction. The scheme (2.7) is competitive with (3.1) only for BFGS direction and small level of noise. It confirms our belief that avoiding $\hat{\sigma}$ in the step size scheme can significantly improve the optimization process. As expected, results clearly demonstrate that our algorithm outperforms corresponding method with SA step sizes (2.2), regardless of the direction, noise levels. Furthermore, the second-order direction, namely BFGS, is significantly better than the negative gradient direction. It also



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

Fig. 4.1. Performance profiles for different values of the noise level

outperforms adaptive algorithm (2.6) which confirms that taking noisy functional values as criterion for adjusting steps can improve the optimization process.

4.2. Testing the algorithms on real data

In this subsection we consider an application of Algorithm 3.1 with $d_k = -G_k$ to a multiple linear regression problem. The multiple linear regression is used to explain the relationship between predictor variables and a response variable by fitting a linear equation to observed data. Let the data set $\{(x_i, y_i)\}_{i=1}^{p}$, with predictor matrix $X \in \mathbb{R}^{p\times d}$ and a response vector $y \in \mathbb{R}^p$ be given. The goal is to minimize the following objective function

$$f(w) = ||y - Xw||_2^2 + \lambda||w||_2^2, \tag{4.2}$$

where $w \in \mathbb{R}^d$ is the model parameter that needs to be estimated, $\lambda \geq 0$ is a regularization parameter and $||\cdot||_2$ is the Euclidean norm. For the value of the regularization parameter we used $\lambda = 0.1$. Note that the objective function (4.2) is convex, therefore Algorithm 3.1 reaches the optimal solution. A stochastic approximations of the objective function and the gradient are calculated using uniformly chosen samples of the training data with the sample size $\lfloor r \cdot p \rfloor, r \in (0, 1)$, where $\lfloor\cdot\rfloor$ denotes the whole-number part. For value of the parameter r we used $r = 0.3$.

In our numerical study, a data set from EUROSTUDENT research conducted in Serbia, Montenegro and Bosnia and Herzegovina in 2014 is used. The EUROSTUDENT project collects comparable data on the social dimension of European higher education. More information can be found at the web page http://www.eurostudent.eu.

The total sample size is $p = 9003$. A multiple regression model is built to assess the impact of potential predictor variables on students' overall satisfaction with their studies. We have considered $d = 5$ predictor variables: social factors, financial factors, external factors, work commitments and institutional factors.

We use the same notation for the algorithms as in Subsection 4.1. The algorithms SAGD (algorithm (1.4)) and MMGD (Algorithm 3.1 with negative gradient direction) are tested with the following parameter specification: $a = 1, A = 1, \alpha = 0.602, \theta = 0.999, m = 10$.

Figure 4.2 reports the performance of the two methods. The vertical axis measures the value of the objective function (cost) and the horizontal axis measures the number of iterations. The

result shows that the MMGD method outperforms SAGD. It can be seen that the cost of MMGD decreases faster, therefore it is more efficient and cheaper in comparison to SAGD.

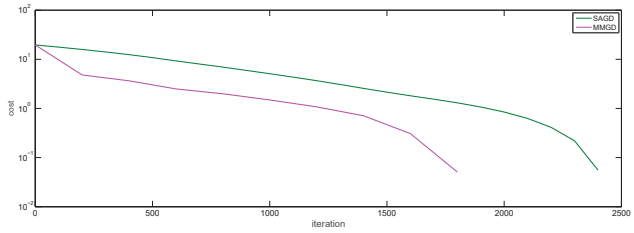


Fig. 4.2. Cost per iteration

5. Conclusions

We have proposed and analyzed a new adaptive step size selection rule for SA algorithms. According to the rule, the step sizes are selected by monitoring previous function values, without knowing or estimating the noise level. We have shown that under common assumption of independent identically distributed continuous random noise with positive pdf, the new adaptive step size sequence has desired SA step sizes convergence property. Numerical results confirmed our expectations for good performance of the proposed SA method with adaptive step sizes.

We believe that the adaptive step size rule can be improved by finding more adequate interval that determines the switching rule among step sizes.

Acknowledgements. We are grateful to the referees, whose suggestions helped us to improve this paper. This research is supported by Ministry of Education, Science and Technology Development of Serbia grant No. 174030 and by Ss. Cyril and Methodius University of Skopje, Macedonia scientific research projects for 2014/2015 academic year.

References

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

[2] J. Blum, Multidimensional stochastic approximation methods, *Ann. Math. Stat.*, **25** (1954), 737-744.

[3] J. Burkhardt, TEST OPT, http://people.sc.fsu.edu/ jburkardt/m_src/test_opt/test_opt.html

[4] R. Byrd, G. Chin, W. Neveitt, J. Nocedal, On the use of stochastic Hessian information in optimization methods for machine learning, *SIAM J. Optim.*, **21**:3 (2011), 977-995.

[5] R. Byrd, G. Chin, J. Nocedal, Y. Wu, Sample size selection in optimization methods for machine learning, *Math. Program.*, **134**:1 (2012), 127-155.

[6] R. Byrd, S. Hansen, J. Nocedal, Singer Y, A stochastic quasi-Newton method for large-scale optimization, *SIAM J. Optim.*, **26**:2 (2016), 1008-1031.

[7] H. Chen, Stochastic Approximation and Its Application, Kluwer Academic Publishers, New York, 2002.

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

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

[10] N. Krejić, Z. Lužanin, I. Stojkovska, A gradient method for unconstrained optimization in noisy environment, *Appl. Numer. Math.*, **70** (2013), 1-21.
[11] N. Krejić, Z. Lužanin, I. Stojkovska, Z. Ovcin, Descent direction method with line search for unconstrained optimization in noisy environment, *Optim. Methods Softw.*, **30**:6 (2015), 1164-1184.
[12] M. Kresoja, Z. Lužanin, I. Stojkovska, Adaptive stochastic approximation algorithm, *Numer. Alg.*, **76**:4 (2017), 917-937.
[13] J. Moré, B. Garbow, K. Hillstrom, Testing unconstrained optimization software, *TOMS*, **7**:1 (1981), 17-41.
[14] W. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, Chapter 6, Stochastic Approximation Methods, John Wiley & Sons, Inc, Hoboken, New Jersey, 2007.
[15] M. Raydan, The Barzilai and Borwein Gradient method for the large scale unconstrained minimization problem, *SIAM J. Optim.*, **7**:1 (1997), 26-33.
[16] H. Robbins, S. Monro, A stochastic approximation method, *Ann. Math. Stat.*, **22** (1951), 400-407.
[17] J. Spall, Adaptive stochastic approximation by the simultaneous perturbarion method, *IEEE AC*, **45**:10 (2000), 1839-1853 .
[18] J. Spall, Introduction to stochastic search and optimization: estimation, simulation and control, John Wiley & Sons, Inc , Hoboken, New Jersey, 2003.
[19] X. Wang, S. Ma, W. Liu, Stochastic quasi-Newton methods for nonconvex stochastic optimization, *SIAM J. Optim.*, **27**:2 (2017), 927-956.
[20] Z. Xu, Y. Dai, A stochastic approximation frame algorithm with adaptive directions, *Numer. Math. Theor. Meth. Appl.*, **1**:4 (2008), 460-474.
[21] Z. Xu, Y. Dai, New stochastic approximation algorithms with adaptive step sizes, *Optim. Lett.*, **6**:8 (2012), 1831-1846.
[22] F. Yousefian, A. Nedic, U. Shanbhag, On Stochastic Gradient and Subgradient Methods with Adaptive Steplength Sequences, *Automatica J. IFAC*, **48**:1 (2012), 56-67.