Adaptive Stochastic Approximation Algorithm

Milena Kresoja * Zorana Lužanin * Irena Stojkovska†

December 19, 2016

Abstract

In this paper stochastic approximation (SA) algorithm with a new adaptive step size scheme is proposed. New adaptive step size scheme uses a fixed number of previous noisy function values to adjust steps at every iteration. The algorithm is formulated for a general descent direction and almost sure convergence is established. The case when negative gradient is chosen as a search direction is also considered. The algorithm is tested on a set of standard test problems. Numerical results show good performance and verify efficiency of the algorithm compared to some of existing algorithms with adaptive step sizes.
Key words. unconstrained optimization, stochastic optimization, stochastic approximation, noisy function, adaptive step size, gradient method, descent direction

1 Introduction

The main objective of this paper is to propose a new method for solving an optimization problem in noisy environment

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

*Department of Mathematics and Informatics, Faculty of Science, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia, e-mail: milena.kresoja@dmi.uns.ac.rs, zorana@dmi.uns.ac.rs. Research supported by Ministry of Education, Science and Technology Development of Serbia grant No. 174030.

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable, possibly nonconvex function bounded below on $\mathbb{R}^n$. We assume that true values of the objective function $f(x)$ and its gradient $\nabla f(x) = g(x)$ are not available but measurable with an error term of stochastic nature. Namely, only noisy measurements of the objective function and gradient are available at all $x \in \mathbb{R}^n$, i.e.

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

where ξ and ε are random variable and random vector, respectively, defined on a probability space $(\Omega, \mathcal{F}, P)$. Also, we will suppose that there is a unique solution $x^* \in \mathbb{R}^n$ of the problem (1).

The notation that we use throughout the paper is

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

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 i.e. the case when the noise-generating process may change with k.

The simple approach for solving problem (1) is introduced in the seminal paper of Robbins and Monro, [1]. The method is called *Stochastic Approximation* (SA) algorithm or RM algorithm. Iterative rule of SA algorithm is inspired by deterministic gradient algorithm. However, instead of a gradient direction it uses available noisy gradient observation G_k at a current iterate x_k, so the next iterate x_{k+1} is calculated as:

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

where a_k is a nonnegative step size. The convergence of SA algorithm is achievable in a stochastic sense under suitable conditions. Robbins and Monro, [1], established a mean square (m.s.) convergence, i.e. $x_k \to x^*$ in m.s., that is $E[\|x_k - x^*\|^2] \to 0$ as $k \to \infty$, while other authors proved the almost sure (a.s.) convergence, i.e. $x_k \to x^*$ a.s. (see [2, 3]). Besides reliance on the noisy gradient evaluations, iterative rule of SA algorithm (4) depends heavily on the choice of the step size sequence. The choice of the step size sequence determines the rate of convergence (see Spall [3] for details). The most used step size sequence is the scaled harmonic sequence $a_k = a/(k+1)$, where $a > 0$. A common generalization of the scaled harmonic sequence is

$a_k = a/(k+1)^\alpha$, where $a > 0$, and $1/2 < \alpha \leq 1$. In addition to harmonic sequences, one of the most used step size length has the form

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

where $a > 0$, $1/2 < \alpha \leq 1$ and $A \geq 0$ is a stability constant which allows taking larger a without risking unstable behaviour in early iterations (for details see [4]). Although ensuring convergence, the step sizes proportional to $1/k$, result in quite slow progress.

Plenty of attempts have been proposed in the literature to improve the SA algorithm. All attempts stand on modifications of the step size and/or search direction selection in (4). Modifications based on step size are discussed in [3, 4, 5, 6, 7, 8, 9], while modification based on search direction are analysed, for example, in [4, 10, 11, 12, 13, 14].

In this paper we focus on a modification of SA algorithm based on adaptive step sizes. The main idea is to adjust the step size in each iteration according to some criterion in order to achieve progress compared to previous iterations. The most popular criterion for step size adjustment is proposed by Kesten in [5]. In this scheme, the step sizes are adjusted according to the frequency of sign changes of the differences between two successive iterations. The same signs indicate that the current iterate is far away from the solution and that larger step size should be taken in the next iterate. On the contrary, the smaller step size should be used when there are frequent sign changes which means that the current iterate is close to the solution. This idea is further considered in [6, 7]. One of the newest schemes, [8], suggests that step size sequence should be a piecewise-constant decreasing function with decrease that occurs when a suitable error threshold is met. This scheme is designed for strongly convex differentiable problems. There are also combinations of SA algorithm with other optimization techniques. For example, the combination of gradient method with line-search and SA step sizes is proposed in [9]. This approach combines line search technique along the negative gradient direction while the iterates are far away from the solution, switching to SA rule when neighbourhood of the solution is reached. The two-phase method is also proposed for general descent direction in [11]. The second-order methods have faster progress while keeping the almost sure convergence. For example, SA algorithm with quasi-Newton direction is successfully applied in [15, 16, 17, 18].

In this paper we propose a general descent direction SA algorithm with a

new adaptive step size scheme. The suggested step size scheme combines the ideas from statistical theory and numerical optimization. A new criterion for adjusting the step sizes tracks fixed number of previous noisy function values and ensures a faster progress of the algorithm when it is expected that larger steps will improve the performance of the algorithm. A guidance for choice of the step size length is suggested. This approach allows the step size sequence to be a sequence of random variables. We also consider separately the case when descent direction is a negative gradient direction. Almost sure convergence is established and algorithms are tested on a set of standard test problems.

The organization of the paper is the following. Section 2 briefly reviews both the gradient method and descent direction method with SA steps. A new step size scheme, algorithm and convergence theory are presented in Section 3. Finally, numerical results are given in Section 4, while conclusions are drawn in Section 5.

2 Preliminaries

For a given initial approximation x_0 of the optimal solution x^*, SA iterative rule is given by (4). The standard convergence conditions for the step size sequence $\{a_k\}$ are

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

The conditions (6) imply that the step size a_k should decay neither too fast, nor too slow. The condition $\sum_k a_k = \infty$ provides that the step size sequence should approach zero sufficiently slow in order to avoid false convergence. The condition $\sum_k a_k^2 < \infty$ provides sufficiently fast decay of step size sequence in order to avoid influence of the noise when the iterates come close to the optimal solution. These conditions are most relevant from user's input point of view.

Let $\{x_k\}$ be a sequence generated by SA method (4). Denoted by $\mathcal{F}_k$ is the $\hat{\sigma}$-algebra generated by $x_0, x_1, \ldots, x_k$. The set of standard assumptions which ensures the convergence of SA algorithm (4) is the following, [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 $\hat{\sigma}$-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 a condition on the shape of $g(x)$, assumption A2 is classical mean-zero condition, while assumption A3 gives restrictions on the magnitude of $g(x)$.

The main convergence result for SA method (4) is the following.

Theorem 2.1 *[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.*

SA method (4) can be extended to a descent direction form. Here we present the descent direction method proposed by Krejić et al. in [11]. 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. For a given initial approximation x_0, iterative rule of the descent direction form of SA method is given by

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

where d_k is a descent direction and $\{a_k\}$ is the sequence of positive gain coefficients.

Convergence of the SA method with descent direction (8) is achievable in a stochastic sense under the set of conditions analogous the classical SA method (4).

Again, let $\{x_k\}$ be a sequence generated by (8) and $\mathcal{F}_k$ is the $\hat{\sigma}$-algebra generated by $x_0, x_1, \ldots, x_k$. We give two new assumptions, [11]:

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

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

A5 There is $c_2 > 0$ such that

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

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

The convergence theorem for the stochastic approximation method with descent direction (8) is the following.

Theorem 2.2 *[11] 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.*

3 New Stochastic Approximation Algorithm

3.1 The Step Size Scheme and the Algorithm

As emphasized in Section 1, the performance of SA algorithm largely depends on the choice of the step size sequence. An adaptive step size selection with a suitable criterion for adoption can significantly enhance its performance. Motivated with this fact, we propose a new adaptive step size scheme that can be successfully applied to both, gradient and general descent direction methods.

While determining the step size a_k, we place emphasis on tracking the previously observed function values F_{k-1}, F_{k-2}, ..., F_{k-m}, for some fixed $m \in \mathbb{N}$, to get insight into whether the objective function is improving. The formulation of our adaptive step size scheme is the following:

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

where $\hat{\sigma} > 0$ and

- $m \in \mathbb{N}$, $m(k) = \min\{k, m\}$, $\theta \in (0, 1)$, $a > 0$, $A \geq 0$, $0.5 < \alpha \leq 1$,

- $s_k = s_{k-1} + I\left\{F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma}\right\}$, for $k = 1, 2, ...,$ and $s_0 = 0$,
- $t_k = t_{k-1} + I\left\{\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma} \leq F_k \leq \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma}\right\}$, for $k = 1, 2, ...,$ and $t_0 = 0$,

where $I(\cdot)$ denotes the indicator function.

As can be seen from (9), in kth iteration we construct an interval

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

based on $m(k)$ previously observed (noisy) function values $F_{k-1}, F_{k-2}, ..., F_{k-m(k)}$. If the observed (noisy) function value in kth iterate F_k, is less than the lower limit of the interval, we consider this scenario to be good, since it represents a "sufficient" decrease of the objective function. In this case we suggest using a larger step size in the next $(k+1)$th iteration. For the larger step, our initial idea was to use a constant full step size, $a_k = a$. Nevertheless, inspired by [8] we chose $a_k = a\theta^{s_k}$, which for large k and large θ still remains large in comparison to step size of the form (5), and we can still obtain properties of the sequence $\{a_k\}$, suitable for convergence analysis. If the observed (noisy) function value in kth iterate F_k is greater than the upper limit of the interval, we reject the current iterate, i.e. zero step size is used, and in this way we block the bad steps, as implemented, for example in [14]. Otherwise, if F_k lies in the interval, we propose a backup step size of the form similar to classical SA step size (5).

The inspiration for intervals J_k is drawn from the interval estimation theory. If an observed function value F_k is considered as an estimate of the optimal function value $f^* = f(x^*)$, than the sequence of the observed function values $F_{k-1}, F_{k-2}, ..., F_{k-m(k)}$ can be considered as its sample of length $m(k)$. So, the interval J_k can be viewed as a confidence-like interval for the expected optimal function value f^*, since it is symmetrical about the sample mean $\frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j}$. And if the next estimate F_k of f^* is in the interval J_k, we decide to proceed with slow but safe steps of the form similar to (5).

Using the adaptive scheme (9) we propose the following algorithm.

ALGORITHM 1 *Mean-Sigma Stochastic Approximation (MS)*

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

Step 1. Direction selection. Choose d_k such that (7) holds.

Step 2. Step size selection. Calculate the noisy function measurement F_k and select the step size a_k according to the criterion (9).

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

A special case of Algorithm 1 is when a negative noisy gradient is chosen as the search direction i.e $d_k = -G_k$.

Algorithm 1 might be more costly in comparison to SA algorithms (4) and (8). It requires an additional measurement of value F_k at each iteration. However, we believe that tracking of the values of the objective function may considerably improve the knowledge about the optimization process. The similar reasoning that using observed function values to accept or reject steps can improve the algorithm's stability is disscused in [3, 14]. This is also a feature by which our algorithm differs from the algorithms with adaptive step sizes in [6, 7]. Thus, an additional measurement at each iteration might be sometimes a good decision, as our numerical results will demonstrate. On the other hand, Algorithm 1 might be a good choice for derivative-free settings, when we must rely on the noisy functional values. In this case, the gradient should be approximated using only functional values, for example with finite differences. We did not consider this case in our numerical experiments, since we suppose that the noisy gradient measurements are known.

Algorithm 1 is defined and its convergence is established (in subsections that follow) for an arbitrary constant $\hat{\sigma} > 0$. But, in practical implementation of the algorithm, as shown in the numerical section, the choice of the constant $\hat{\sigma}$ is closely related to the noise level. Namely, it can be easily shown that in the case of an independent and identically distributed "white noise" with variance σ^2, i.e. $E(\xi_k) = 0$ and $Var(\xi_k) = \sigma^2$, for all k, the mean-square error (MSE) of the function value estimator F_k of the optimal value f^* is equal to $\sigma^2 + (f_k - f^*)^2$, where $f_k = f(x_k)$ is true function value at x_k. Now, since the variance of the sampling distribution of F_k is often approximated reasonably well by MSE of F_k (see [19]), it is justified to relate the noise level σ to the constant $\hat{\sigma}$ in the interval J_k.

3.2 Properties of the Adaptive Step Size Sequence

In this subsection we analyze the step size sequence $\{a_k\}$ generated by the adaptive step size scheme (9). The adaptive step size scheme (9) forms a sequence of random variables. In order to establish a convergence of the proposed algorithm we will show that the sequence $\{a_k\}$ satisfies the conditions (6) a.s. under reasonable assumptions on the noise terms.

By using the introduced notations (3), we impose the following conditions on the noise terms ξ_k:

$$\begin{aligned}&\xi_k, k = 0, 1, 2, ... \text{ are i.i.d. continuous random variables with a common}\\ &\text{probability density function (pdf) } p(x) > 0 \text{ a.s. for all } x \in \mathbb{R}\end{aligned} \tag{10}$$

The conditions (10) do not have any real restrictions, since the noise usually occurs independently. An example of the noise that satisfies conditions (10) is independent identically distributed normal Gaussian noise which is also used in our numerical experiments.

We start analysing the step size sequence $\{a_k\}$ by focusing on the following event

$$A_k = \left\{a_{k-1} = a_{k-2} = \ldots = a_{k-m(k)} = 0\right\}. \tag{11}$$

Realization of the event A_k means that $m(k)$ consecutive zero steps occurred.

Lemma 3.1 *Let the step sizes a_k be defined by (9). If the noise terms ξ_k satisfy the conditions (10), then for $k = 1, 2, \ldots$, the following inequality holds*

$$P(A_k) > 0, \tag{12}$$

where A_k is the event defined by (11).

Proof. Lemma states that $m(k)$ consecutive zero steps occur with nonzero probability. We will prove it by assuming the contrary, i.e. let us assume that there exists k such that

$$0 = P(A_k) = P(F_{k-i} > \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, \ldots, m(k)). \tag{13}$$

Let us consider the events $\left\{F_{k-i} > \max_{1\leq j\leq m(k)} F_{k-i-j} + \hat{\sigma}\right\}$, $i = 1, 2, ..., m(k)$.

Obviously,

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

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

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

which further implies

$$P(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, ..., m(k))$$
$$\le P(F_{k-i} > \frac{1}{m(k)}\sum_{j=1}^{m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, ..., m(k)). \tag{14}$$

Relations (13) and (14) imply

$$P(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, \ldots, m(k)) = 0. \tag{15}$$

Let us now define a *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 $B^k_{\frac{\delta}{2}}$ the event

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

Now, we chose $\delta > 0$ such that

$$P(B^k_{\frac{\delta}{2}}) > 0. \tag{16}$$

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

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

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

Now,

$$\begin{aligned} 0 &= P\left(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma}, i = 1, 2, \ldots, m(k)\right) \\ &\geq P\left(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma}, i = 1, 2, \ldots, m(k) \big| B^k_{\frac{\delta}{2}}\right) P(B^k_{\frac{\delta}{2}}). \end{aligned} \quad (17)$$

So, (16) and (17) imply

$$P\left(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, \ldots, m(k) \big| B^k_{\frac{\delta}{2}}\right) = 0. \quad (18)$$

Under the realization of the event $B^k_{\frac{\delta}{2}}$, it can be shown that

$$f_{k-i} - \delta < f_{k-j} < f_{k-i} + \delta, \quad (19)$$

for all $i, j = 1, 2, ..., 2m(k)$. Now, using (19), under the realization of the event $B^k_{\frac{\delta}{2}}$, the inequality

$$\xi_{k-i} > \max_{1\le j\le m(k)} \xi_{k-i-j} + \hat{\sigma} + \delta$$

implies

$$F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma},$$

and this is true for any $i = 1, 2, \ldots, m(k)$. Therefore

$$\begin{aligned} 0 &= P\left(F_{k-i} > \max_{1\le j\le m(k)} F_{k-i-j} + \hat{\sigma},\ i = 1, 2, \ldots, m(k) \big| B^k_{\frac{\delta}{2}}\right) \\ &\geq P\left(\xi_{k-i} > \max_{1\le j\le m(k)} \xi_{k-i-j} + \hat{\sigma} + \delta,\ i = 1, 2, \ldots, m(k) \big| B^k_{\frac{\delta}{2}}\right) \\ &= P\left(\xi_{k-i} > \max_{1\le j\le m(k)} \xi_{k-i-j} + \hat{\sigma} + \delta,\ i = 1, 2, \ldots, m(k)\right), \end{aligned} \quad (20)$$

since the last conditional probability is independent of the condition. Relation (20) implies

$$P\left(\xi_{k-i} > \max_{1\le j\le m(k)} \xi_{k-i-j} + \hat{\sigma} + \delta,\ i = 1, 2, \ldots, m(k)\right) = 0. \quad (21)$$

Now,

$$\begin{aligned} 0 &= P\left(\xi_{k-i} > \max_{1\le j\le m(k)} \xi_{k-i-j} + \hat{\sigma} + \delta,\ i = 1, 2, \ldots, m(k)\right) \\ &= P\left(\xi_{k-i} > \xi_{k-i-j} + \hat{\sigma} + \delta,\ i, j = 1, 2, \ldots, m(k)\right) \\ &\geq P\left(\xi_{k-1} > \xi_{k-2} + \hat{\sigma} + \delta > \xi_{k-3} + 2(\hat{\sigma} + \delta) > \ldots > \xi_{k-2m(k)} + (2m(k) - 1)(\hat{\sigma} + \delta)\right) \\ &= I(\hat{\sigma} + \delta). \end{aligned} \tag{22}$$

On the other hand,

$$\begin{aligned} I(\hat{\sigma} + \delta) &= \int_{-\infty}^{\infty} p(x_{k-1}) \int_{-\infty}^{x_{k-1}-(\hat{\sigma}+\delta)} p(x_{k-2}) \ \cdots \\ &\quad \int_{-\infty}^{x_{k-2m(k)+1}-(2m(k)-1)(\hat{\sigma}+\delta)} p(x_{k-2m(k)})\, dx_{k-1}dx_{k-2}...dx_{k-2m(k)} > 0 \end{aligned}$$

almost surely for all $\delta > 0$, since $p(x) > 0$ a.s. by condition (10), and $I(\delta)$ is a decreasing function, which is in contradiction with (22). This implies that $P(A_k) > 0$ for all k. ∎

Now, when we know that in each iteration, $m(k)$ consecutive zero steps may occur with non zero probability, we can state the following lemma for probability distribution of the step sizes a_k.

Lemma 3.2 *Let the step sizes a_k be defined by (9). If the noise terms ξ_k satisfy the conditions (10), then for all $k = 1, 2, \ldots$*

$$P(a_k = 0|A_k) > 0, \tag{23}$$

$$P(a_k = a\theta^{s_k}|A_k) > 0, \tag{24}$$

and

$$P(a_k = \frac{a}{(t_k + 1 + A)^{\alpha}}|A_k) > 0, \tag{25}$$

where A_k is the event defined by (11). Moreover, for all $k = 1, 2, ...$

$$P(a_k = 0) > 0, \tag{26}$$

$$P(a_k = a\theta^{s_k}) > 0, \tag{27}$$

and

$$P(a_k = \frac{a}{(t_k + 1 + A)^{\alpha}}) > 0. \tag{28}$$

Proof. First note that the conditional probabilities (23)-(25) are well defined because of Lemma 3.1. Then, under the realization of the event A_k we have $f_k = \frac{1}{m(k)} \sum_{j=1}^{m(k)} f_{k-j}$.

Let us start with proving the first probability (23). According to step size rule (9) we have

$$\begin{aligned} P(a_k = 0|A_k) &= P(F_k > \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma}|A_k) \\ &= P(f_k + \xi_k > \frac{1}{m(k)} \sum_{j=1}^{m(k)} (f_{k-j} + \xi_{k-j}) + \hat{\sigma}|A_k) \\ &= P(\xi_k > \frac{1}{m(k)} \sum_{j=1}^{m(k)} \xi_{k-j} + \hat{\sigma}|A_k) \\ &= P(\xi_k - \frac{1}{m(k)} \sum_{j=1}^{m(k)} \xi_{k-j} > \hat{\sigma}), \end{aligned} \tag{29}$$

since the conditional probability is independent of the condition. Let us denote variable Y_k by

$$Y_k = \xi_k - \frac{1}{m(k)} \sum_{j=1}^{m(k)} \xi_{k-j}, \tag{30}$$

and by $p_{Y_k}(\cdot)$ its pdf. We can think of Y_k as a difference of two random variables, ξ_k with pdf $p(\cdot)$ and $Z_{k,m(k)} = \frac{1}{m(k)} \sum_{j=1}^{m(k)} \xi_{k-j}$ with pdf $p_{k,m(k)}(\cdot)$. By the convolution formula for two independent random variables X and Y, the pdf of their sum $X + Y$ is

$$p_{X+Y}(z) = \int_{-\infty}^{\infty} p_Y(z - t) p_X(t) dt. \tag{31}$$

where $p_X(\cdot)$ is pdf of X, and $p_Y(\cdot)$ is pdf of Y. Now, using (31) we can derive recursively the distribution of the random variable $Z_{k,m(k)}$, since ξ_k are all independent random variables, by condition (10). The derived probability density function $p_{k,m(k)}(\cdot)$ is always positive because it only depends on $p(\cdot)$ which is, by condition (10), always positive. Now, the pdf of Y_k is

$$p_{Y_k}(y) = \int_{-\infty}^{\infty} p(t) p_{k,m(k)}(y - t) dt, \tag{32}$$

and it is always positive, since $p(\cdot)$ and $p_{k,m(k)}(\cdot)$ are always positive. If we implement these findings in (29), we will have

$$P(a_k = 0|A_k) = P(Y_k > \hat{\sigma}) = \int_{\hat{\sigma}}^{\infty} p_{Y_k}(y)dy > 0. \tag{33}$$

Similarly, we have

$$P(a_k = a\theta^{s_k}|A_k) = P(Y_k < -\hat{\sigma}) = \int_{-\infty}^{-\hat{\sigma}} p_{Y_k}(y)dy > 0 \tag{34}$$

and

$$P(a_k = \frac{a}{(t_k+1+A)^{\alpha}}|A_k) = P(-\hat{\sigma} \leq Y_k \leq \hat{\sigma}) = \int_{-\hat{\sigma}}^{\hat{\sigma}} p_{Y_k}(y)dy > 0, \tag{35}$$

since $\hat{\sigma} > 0$. Additionally, from Lemma 3.1 and (33)-(35), for all $k = 1, 2, ...$ we have

$$P(a_k = 0) \geq P(a_k = 0|A_k) \cdot P(A_k) > 0, \tag{36}$$

$$P(a_k = a\theta^{s_k}) \geq P(a_k = a\theta^{s_k}|A_k) \cdot P(A_k) > 0, \tag{37}$$

and

$$P(a_k = \frac{a}{(t_k+1+A)^{\alpha}}) \geq P(a_k = \frac{a}{(t_k+1+A)^{\alpha}}|A_k) \cdot P(A_k) > 0, \tag{38}$$

which completes the proof. ∎

Previous Lemma 3.2 leads to important results which are stated below.

Lemma 3.3 *Let the step sizes a_k be defined by (9). If the noise terms ξ_k satisfy the condition (10), then almost surely there are infinitely many steps $a_k = \frac{a}{(t_k+1+A)^{\alpha}}$ and infinitely many steps $a_k = a\theta^{s_k}$.*

Proof.
Let us first consider the sequence of events $T_k = \left\{a_k = \frac{a}{(t_k+1+A)^{\alpha}}\right\}$, $k = 1, 2, 3, \ldots$. Define $\{T_k \text{ i.o.}\}$ as the event that an infinite number of events T_k, $k = 1, 2, 3, \ldots$ occur. The *i.o.* stands for *infinitely often*. We will show that event $\{T_k \text{ i.o.}\}$ occurs almost surely, i.e.

$$P(\{T_k \text{ i.o.}\}) = P(\{w|w \in T_k \text{ for infinite many } k \in \{1, 2, 3, ...\}\}) = 1. \tag{39}$$

Let us consider the subsequence $\left\{T_{k(m+1)}\right\}_k$ of the sequence $\{T_k\}_k$. It is a sequence of independent events, because they depend on different random variables ξ_k, which are independent by (10). The event $\{T_{k(m+1)} \text{ i.o.}\}$ is a member of the tail σ-algebra $\bigcap_{k=1}^{\infty}\left\{\sigma(T_{n(m+1)}), n \geq k\right\}$. Therefore, we can apply Kolmogorov $0-1$ law which states that for a sequence of independent events, probability of any tail event is 0 or 1, [20]. According to Kolmogorov $0-1$ law,

$$P(\{T_{k(m+1)} \text{ i.o.}\}) \in \{0,1\}. \tag{40}$$

Let us assume that

$$P(\{T_{k(m+1)} \text{ i.o.}\}) = 0. \tag{41}$$

Because of the inclusion

$$\bigcap_{k=1}^{\infty} T_{k(m+1)} \subseteq \{T_{k(m+1)} \text{ i.o.}\},$$

we have that

$$P\left(\bigcap_{k=1}^{\infty} T_{k(m+1)}\right) \leq P(\{T_{k(m+1)} \text{ i.o.}\}),$$

that together with (41) imply

$$P\left(\bigcap_{k=1}^{\infty} T_{k(m+1)}\right) = 0. \tag{42}$$

As we mentioned before, $T_{k(m+1)}$, $k = 1, 2, 3, ...$ are independent events, so (42) is equivalent to

$$\prod_{k=1}^{\infty} P(T_{k(m+1)}) = 0, \tag{43}$$

which implies that there exists $k_0 \in \mathbb{N}$ such that $P(T_{k_0(m+1)}) = 0$ i.e. $P(a_{k_0} = \frac{a}{(t_{k_0}+1+A)^{\alpha}}) = 0$ which is in contradiction to (28) from Lemma 3.2. Therefore,

$$P(\{T_{k(m+1)} \text{ i.o.}\}) > 0. \tag{44}$$

The relation (44) together with (40) implies

$$P(\{T_{k(m+1)} \text{ i.o.}\}) = 1. \tag{45}$$

Now, because of the inclusion

$$\{T_{k(m+1)} \text{ i.o.}\} \subseteq \{T_k \text{ i.o.}\}, \tag{46}$$

we have that

$$P(\{T_{k(m+1)} \text{ i.o.}\}) \leq P(\{T_k \text{ i.o.}\}), \tag{47}$$

that together with (45) imply (39), i.e. almost surely there are infinitely many steps $a_k = \frac{a}{(t_k+1+A)^\alpha}$. Analogously, we can show that almost surely there are infinitely many steps $a_k = a\theta^{s_k}$, which completes the proof. ∎

Remark 3.1 *As a consequence of Lemma 3.3 we have that almost surely infinitely many successive steps $a_k = 0$ cannot occur, since almost surely there are infinitely many non zero steps. This finding will help us during the practical implementation of the Algorithm 1. We can impose a correction condition and constrain the number of consecutive zero step in the following way. If there are some predefined number of successive steps $a_k = 0$, then in the next iteration we are going to take a non zero safe step of the form (5).*

Lemma 3.3 helps us to show that the step size sequence $\{a_k\}$ satisfies conditions (6) a.s.

Theorem 3.1 *If the noise terms ξ_k satisfy the condition (10), then the step size sequence $\{a_k\}$, defined by (9), satisfies the conditions (6) a.s.*

Proof. If we denote by $C = \{k | F_k < \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma}\}$ and $D = \{k | \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} - \hat{\sigma} \leq F_k \leq \frac{1}{m(k)} \sum_{j=1}^{m(k)} F_{k-j} + \hat{\sigma}\}$, then by the definition of the sequence $\{a_k\}$, the equation (9), we have

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

and

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

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 3.3. So, the step size sequence $\{a_k\}$ satisfies the conditions (6) a.s. ∎

3.3 Convergence Analysis

In this subsection, we establish the convergence of the Algorithm 1. We discuss separately a case when direction is a negative gradient.

The previously mentioned SA convergence theorems, Theorem 2.1 and Theorem 2.2, 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 the following. The steps a_k are $\mathcal{F}_k$-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. Therefore, we are not allowed to use information from $(k+1)$th iteration to compute a_k, similar to the assumption in [21]. We also need to assume that conditions (6) are satisfied almost surely (a.s.). Under these additional assumptions, the SA convergence theorems, Theorem 2.1 and Theorem 2.2, also hold when step sizes a_k are stochastic.

Theorem 3.1 ensures that the step sizes a_k generated by Algorithm 1 satisfies the conditions (6) almost surely. Due to the convergence theorem for descent direction method with SA step sizes, Theorem 2.2, it follows that assumptions A2-A5 and Theorem 3.1 ensure almost surely convergence of Algorithm 1. Thus, we have the following convergence result for the method with adaptive step sizes proposed in Algorithm 1.

Theorem 3.2 *Assume that A2-A5 hold. Let $\{x_k\}$ be a sequence generated by Algorithm 1, where the noise terms ξ_k satisfy the condition (10). 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 2.1 for stochastic step sizes a_k, 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 1 with $d_k = -G_k$, where the noise terms ξ_k satisfy the condition (10). Then the sequence $\{x_k\}$ converges to x^* a.s. for an arbitrary initial approximation x_0.*

4 Numerical Results

In this section, numerical results are presented. A gradient version of Algorithm 1 is compared to SA algorithm (4) and algorithms with adaptive steps

from [6] and [7]. A descent direction form of Algorithm 1 is compared to SA algorithm with descent direction (8).

Testing is performed on the 18 test problems selected from [22] and [23]. All problems have the form of nonlinear least squares

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

The list of problems, their dimensions n and the initial approximations x_0 are given in Table 1. The problems are transformed into noisy problems by adding a normal distributed noise of the form

$$\xi \sim \mathcal{N}(0, \sigma^2) \quad \text{and} \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I_{n\times n}), \tag{48}$$

to the function and gradient evaluations respectively, where $\sigma > 0$ is the noise level and $I_{n\times n}$ is the identity matrix. Three different values for σ are tested, $\sigma = 0.01, 0.4, 1$.

The noisy functional and gradient values are calculated using the arithmetic mean with sample size p. i.e.

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

where $\{\xi_k^i\}$ and $\{\varepsilon_k^i\}$ are i.i.d. samples and p is some small positive integer. All tests are performed with sample size $p = 3$.

The values of the parameters a, A and α used in step sizes (5) and (9) are shown in Table 2. Some of them are taken form [6] and [14], while the others are derived as the most suitable choice for the underlying problem.

The descent direction forms of Algorithm 1 and SA algorithm (8) are tested using a quasi-Newton direction. In particular, we use BFGS direction $d_k = -B_k^{-1}G_k$, with the update formula

$$B_{k+1} = B_k - \frac{B_k\delta_k\delta_k^T B_k}{\delta_k^T B_k \delta_k} + \frac{\Delta_k\Delta_k^T}{\Delta_k\delta_k}, \tag{49}$$

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. This is already successfully tested in [11, 17, 18, 24].

Table 1: Test problems.

No	Problem	n	x_0
1	The Gaussian function	3	$(4/10, 1, 0)$
2	The Box 3-dimensional function	3	$(0, 10, 5)$
3	The variably dimensioned function	4	$(3/4, 2/4, 1/4, 0)$
4	The Watson function	4	$(0, 0, 0, 0)$
5	The Penalty Function 1	10	$(1, 1, \ldots, 1)$
6	The Penalty Function 2	4	$(1/2, 1/2, 1/2, 1/2)$
7	The Trigonometric Function	10	$(1/10, 1/10, \ldots, 1/10)$
8	The Beale Function	2	$(1, 1)$
9	The Chebyquad Function	10	$(5/11, 10/11 \ldots, 50/11)$
10	The Gregory and Karney Tridiagonal Matrix Function	4	$(0, 0, 0, 0)$
11	The Hilbert Matrix Function	4	$(1, 1, 1, 1)$
12	The De Jong Function 1	3	$(-5.12, 0, 5.12)$
13	The Branin RCOS Function	2	$(-1, 1)$
14	The Colville Polynomial	4	$(1/2, 1, -1/2, -1)$
15	The Powell 3D Function	3	$(0, 1, 2)$
16	The Himmelblau function	2	$(-1.3, 2.7)$
17	Strictly Convex 1	10	$(1/10, 2/10, \ldots, 1)$
18	Strictly Convex 2	10	$(1, 1, \ldots, 1)$

Each test consists of $N = 50$ independent runs starting from the same initial point. Algorithms stop if $||G_k|| \leq c$, where $c = \min\{\sqrt{n}\sigma, 1\}$, or when the maximal number of $200n$ function evaluations are reached, with each gradient evaluation counted as n function evaluations. That is, the algorithms stop if either a stationary point in a stochastic sense is reached or the maximal number of function evaluations is used. Like in [11], runs are classified into three categories successful (convergent), partially successful runs and unsuccessful (divergent) runs. A run is considered successful if a method stops due to $||G_k|| \leq c$. The number of successful runs is denoted by $Nconv$. If $||G_k|| > 200\sqrt{n}$, run is unsuccessful, i.e. divergence is declared. The number of divergent runs is denoted by $Ndiv$. Finally, the runs that stop due to exhausting the maximal number of allowed function evaluations are considered partially successful and their number is denoted by $Npar$.

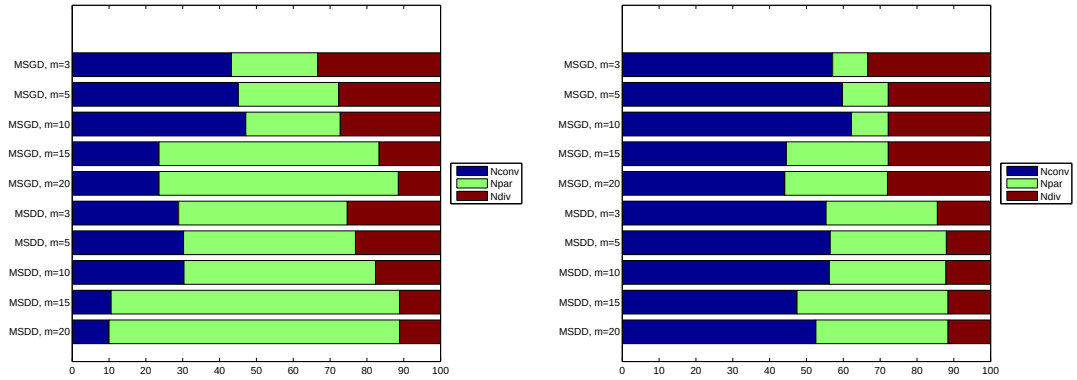
Table 2: The initialization of the parameters.

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

We explore the performance and sensitivity of Algorithm 1 to the parameters θ, m and $\hat{\sigma}$ used in the step size scheme (9). We report results for larger values of the parameter θ, that are $\theta = 0.75$ and $\theta = 0.99$, since our initial hypothesis was that a larger θ would improve performance of the algorithm allowing bigger steps when a "sufficient" decrease in functional value is observed. Next, we consider five different values for $m = 3, 5, 10, 15, 20$. For the parameter $\hat{\sigma}$ in the step size scheme (9) we use the noise level σ given by (48), which, as we explained earlier, is closely related to the variance of the sampling distribution of the estimator F_k of the optimal value f^*.

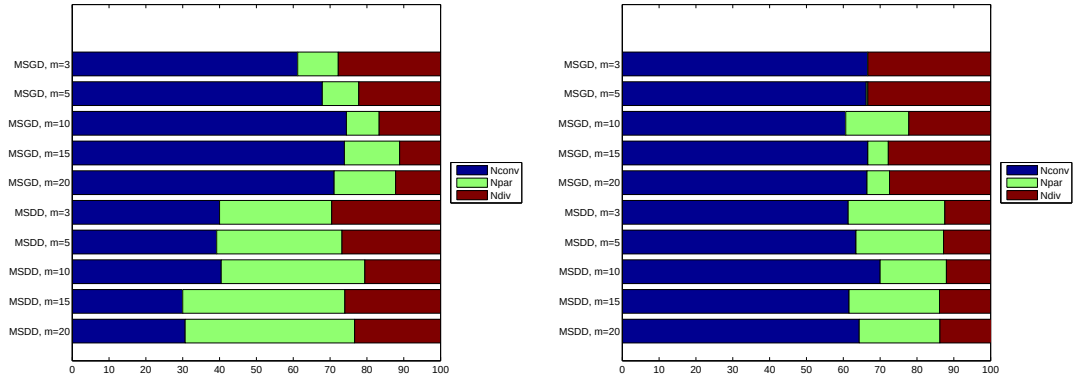
Consecutive zero steps that can occur during the implementation of Algorithm 1 may lead to no progress of the algorithm. As an additional implementation issue, based on Remark 3.1, we constrain the number of consecutive zero steps. The following correction is applied. If the number of consecutive

zero steps is greater than some predetermined number m_{corr}, in the next iteration we use $a_k = \frac{a}{(t_k+1+A)^\alpha}$ as a step size. Empirically, we decided it is best to use $m_{corr} = m + 1$ as a correction value.



a. $\theta = 0.75$ b. $\theta = 0.99$

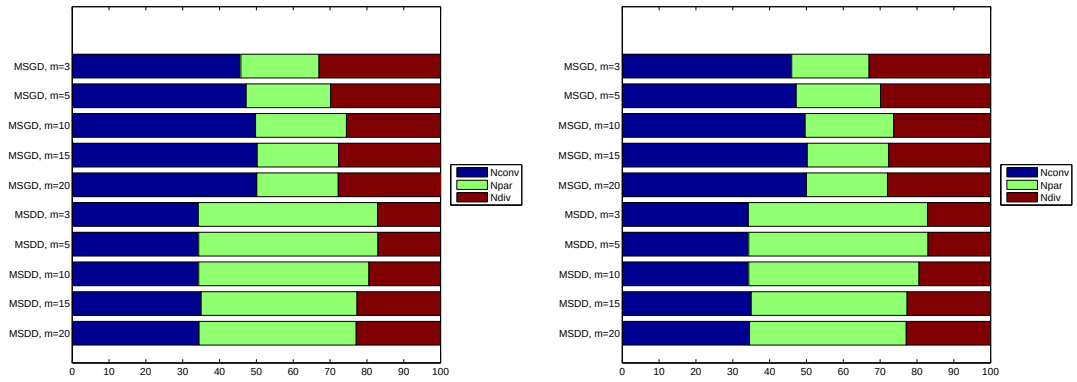
Figure 1: Algorithm 1: Percentage of successful, partially successful and divergent runs, $\sigma = 0.01$



a. $\theta = 0.75$ b. $\theta = 0.99$

Figure 2: Algorithm 1: Percentage of successful, partially successful and divergent runs, $\sigma = 0.4$

Overviews of successful, partially successful and unsuccessful runs of the Algorithm 1 for the noise levels $\sigma = 0.01$, $\sigma = 0.4$ and $\sigma = 1$ are given in Figure 1, Figure 2, and Figure 3 respectively. All three figures display results of testing Algorithm 1 with the negative gradient direction (MSGD) and with



a. $\theta = 0.75$ b. $\theta = 0.99$

Figure 3: Algorithm 1: Percentage of successful, partially successful and divergent runs, $\sigma = 1$

the BFGS direction (MSDD) for $\theta = 0.75, 0.99$ and $m = 3, 5, 10, 15, 20$. The results show that the performance of Algorithm 1 is sensitive to the choice of all three parameters θ, m and $\hat{\sigma}$, as well as being sensitive to the chosen direction. In almost all cases, regardless of the chosen direction, the noise level σ and parameter m, results show that higher values of θ lead to a smaller number of divergent runs and higher number of convergent runs. This is in conformance with our discussion that larger steps can improve the performance of the algorithm when a "sufficient" decrease in current functional value is observed. Similarly, in almost all cases higher m leads to smaller number of divergent runs and higher number of convergent runs. However, due to the fact that average value is sensitive to extreme values we do not recommend taking too large values for m. The most suitable value, derived empirically, is $m = 10$. It was expected that the lower noise level should give better performance, but that did not happen (see Figure 1 for $\theta = 0.99$ and Figure 2 for $\theta = 0.99$), probably due to the very small interval for $\sigma = 0.01$ in the adaptive step size scheme (9) that accepts safe steps of the form (5). This is verified by additional testing which allowed noise level σ and parameter $\hat{\sigma}$ to differ. Three different parameters $\hat{\sigma} = 0.1, 0.3, 0.6$ are tested for all three noise levels $\sigma = 0.01, 0.4, 1$. The numerical results are available at the web page `http://irenastojkovska.weebly.com/uploads/5/8/2/0/58202701/adaptivesa_add2.pdf`.

Finally, we compare Algorithm 1 which uses the negative gradient direc-

tion and the BFGS direction with other relevant adaptive algorithms. We present results for the following 6 algorithms:

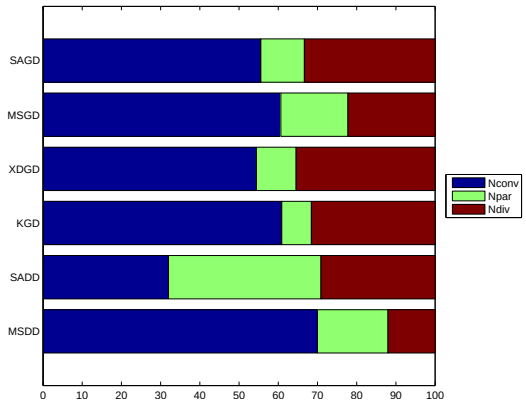
- SAGD - Algorithm (4) with SA step sizes (5)
- MSGD - Algorithm 1 with $d_k = -G_k$, $\theta = 0.99$ and $m = 10$
- XDGD - adaptive step size algorithm from [7]
- KGD - adaptive step size algorithm from [6]
- SADD - Algorithm (8) with BFGS direction and SA step sizes (5)
- MSDD - Algorithm 1 with BFGS direction, $\theta = 0.99$ and $m = 10$

Note that the first 4 algorithms use negative gradient directions. In Figure 4 and Figure 5 we report the percentages of successful, partially successful and unsuccessful runs, together with the performance profiles for noise levels $\sigma = 0.4$ and $\sigma = 1$ respectively. As a performance measure we use the number of function evaluations needed in successful and partially successful runs i.e.

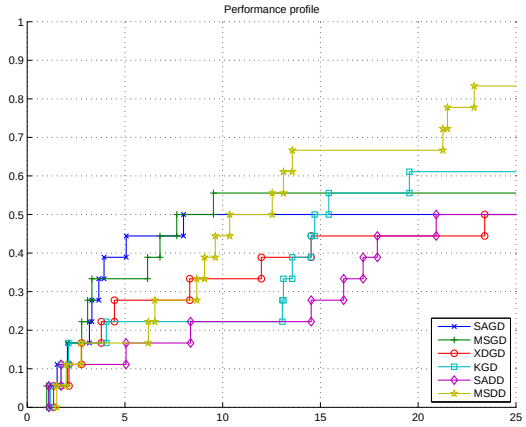
$$\pi_{ij} = \frac{1}{|Ncon_{ij} \bigcup Npar_{ij}|} \sum_{r \in Ncon_{ij} \bigcup Npar_{ij}} \frac{fcalc_{ij}^r}{n_j},$$

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, where $i = 1, \cdots, 7,\ j = 1, \cdots, 18,\ r = 1, \cdots, 50$.

The results demonstrate that Algorithm 1 has smaller number of divergent runs regardless of the chosen direction and noise level. Kesten's algorithm (KGD) is competitive with the gradient form of Algorithm 1 (MSGD) in the number of successful runs but fails when it comes to the number of divergent runs. Our algorithms with BFGS direction (MSDD) is significantly better than the corresponding SA algorithm (8) with BFGS direction (SADD) for both noise levels. We can conclude that Algorithm 1 is competitive in comparison with the existing algorithms with adaptive step sizes, while being more successful in decreasing the number of the divergent runs. Good performance of Algorithm 1 is also confirmed with performance profiles.

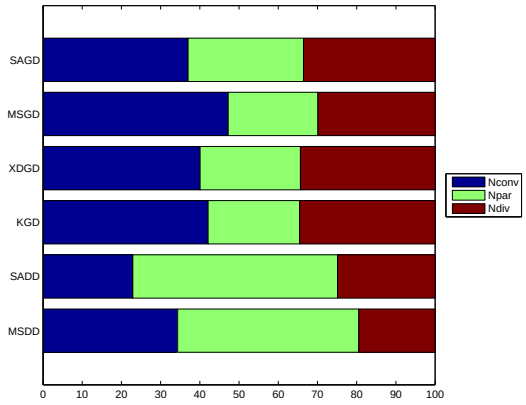


a. Percentage of successful, partially successful and divergent runs

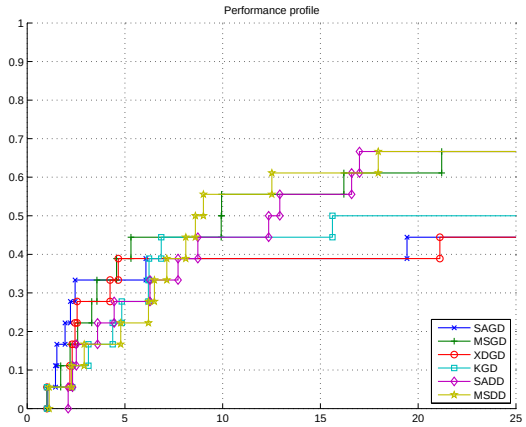


b. Performance profile

Figure 4: Comparison of the algorithms, $\sigma = 0.4$



a. Percentage of successful, partially successful and divergent runs



b. Performance profile

Figure 5: Comparison of the algorithms $\sigma = 1$

5 Conclusions

In this paper we have proposed a new adaptive step size scheme for the stochastic approximation (SA) algorithms based on the tracking previously noisy function values. According to the proposed scheme, the larger step size

in the next iterate is adopted if a "sufficient" decrease in the current functional value is observed. Under a non restrictive assumption of independent identically distributed continuous random noise with a positive pdf, the generated step size sequence has the desired SA step size property that ensures the almost sure convergence of SA methods for both gradient and descent direction. Numerical results verify better performance of the proposed SA algorithm with adaptive step sizes compared to the existing algorithms with adaptive step sizes.

In the future, we might focus our attention on analyzing the convergence of the proposed methods when $\theta = 1$. It will be challenging to analyze convergence in a more general case of state dependent noise and with no restrictions to its pdf, since we obtained good numerical results in those cases too. Finally, it will also be interesting to introduce variability in constants m and $\hat{\sigma}$ used in the adaptive step size scheme (9).

References

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

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

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

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

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

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

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

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

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

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

[11] Krejic, N., Luzanin, 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), DOI: 10.1080/10556788.2015.1025403

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

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

[14] Xu, Z., Dai, Y. H.: A stochastic approximation frame algorithm with adaptive directions, Numer. Math. Theor. Meth. Appl. 1(4), 460-474 (2008)

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

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

[17] Byrd, R. H., Hansen, S. L., Nocedal, J., Singer, Y.: A Stochastic Quasi-Newton Method for Large-Scale Optimization, SIAM J. Optim., 26(2), 10081031 (2016)

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

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

[20] Durrett, R.: Probability: Theory and Examples, Second edition, Duxbury Press, Belmont, CA (1995)

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

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

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

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