fractal and fractional

Article

Adaptive Stochastic Gradient Descent Method for Convex and Non-Convex Optimization

Ruijuan Chen [1], Xiaoquan Tang [2] and Xiuting Li [3,*]

[1] Research Center of Nonlinear Science, School of Mathematical and Physical Sciences, Wuhan Textile University, Wuhan 430200, China
[2] School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen 518172, China
[3] College of Science, Huazhong Agricultural University, Wuhan 430070, China
* Correspondence: xtingli@mail.hzau.edu.cn

Abstract: Stochastic gradient descent is the method of choice for solving large-scale optimization problems in machine learning. However, the question of how to effectively select the step-sizes in stochastic gradient descent methods is challenging, and can greatly influence the performance of stochastic gradient descent algorithms. In this paper, we propose a class of faster adaptive gradient descent methods, named AdaSGD, for solving both the convex and non-convex optimization problems. The novelty of this method is that it uses a new adaptive step size that depends on the expectation of the past stochastic gradient and its second moment, which makes it efficient and scalable for big data and high parameter dimensions. We show theoretically that the proposed AdaSGD algorithm has a convergence rate of $O(1/T)$ in both convex and non-convex settings, where T is the maximum number of iterations. In addition, we extend the proposed AdaSGD to the case of momentum and obtain the same convergence rate for AdaSGD with momentum. To illustrate our theoretical results, several numerical experiments for solving problems arising in machine learning are made to verify the promise of the proposed method.

Keywords: stochastic gradient descent; adaptive step size; convex optimization; non-convex optimization

Citation: Chen, R.; Tang, X.; Li, X. Adaptive Stochastic Gradient Descent Method for Convex and Non-Convex Optimization. *Fractal Fract.* **2022**, *6*, 709. https://doi.org/10.3390/fractalfract6120709

Academic Editors: Ricardo Almeida, Dongfang Li and Xiaoli Chen

Received: 10 August 2022
Accepted: 23 November 2022
Published: 29 November 2022

1. Introduction

Optimization based on stochastic gradients is of central practical significance in many scientific and engineering fields. Many problems in these areas can be reduced to optimization problems of some scalar parameterized objective function for which parameters need to be maximized or minimized. Recent years have witnessed the great success of machine learning, especially deep learning, in many fields, including computer vision, speech processing, and natural language processing. For many machine learning tasks, a critical and challenging problem is to design optimization algorithms to train neural network models. If the objective function is differentiable, stochastic gradient descent (SGD) is an efficient and effective optimization method that plays a central role in many machine learning successes. The SGD algorithm can be traced back to Robbins and Monro [1], who stated that the classical convergence analysis depends on the decreasing positive learning rate condition. Stochastic approximation methods have been widely studied in various areas of the literature [2–4], mainly focusing on the convergence of algorithms in different environments.

In recent years, the convergence speed of standard SGD has been greatly improved, and a number of methods to reduce variance have been adopted, such as vanilla SGD in the non-convex case [5]. However, vanilla SGD is too sensitive to the learning rate, making it difficult to adjust the appropriate learning rate, and its convergence performance is poor. There have been many attempts to achieve easily tunable learning rates and improve SGD performance, For example, in the case of the smooth and strongly convex objective function, the variance of stochastic gradient decrease [6–9], adaptive learning rate [10–16], averaging [17], momentum acceleration mechanism [18–21], and the Powerball method [22] are used, and a better self optimization control method is proposed using fractional-order Gaussian noise [23]. The most promising variance reduction technique is the stochastic variance reduction gradient (SVRG) [8,9]. In fact, these stochastic methods need to store and use the full batch of past gradients in order to progressively reduce the variance of the stochastic gradient estimator. For stochastic optimization problems, the number of training samples is usually large; consequently, the algorithm can be difficult to implement if the storage space is limited. Therefore, adaptive learning rate and momentum mechanisms are more suitable for stochastic optimization problems than variance reduction.

In addition to the classical optimization algorithms, several other popular stochastic optimization algorithms can be found in the current literature, for example, genetic algorithms, which are inspired by biological evolution [24], particle swarm optimization derived from the natural behavior of clusters [25,26], and the most recent dynamic stochastic fractal search optimization algorithm based on the adaptive strategy of fuzzy logic for diffusion parameters [27]. However, because heuristic algorithms are proposed based on experience without a theoretical basis, they lack a unified and complete theoretical framework. In addition, due to the use of non-deterministic polynomial theory, global optimality cannot be guaranteed when using the heuristic polynomial approach.

Adaptive step sizes have a long history in convex settings. They were first proposed in the online learning literature [28] and later applied to the random learning literature [12]. In a recent study, an adaptive projection gradient algorithm has been proposed for a special nonlinear fractional optimization problem with an objective function that is smooth convex in the numerator and smooth concave in the denominator [29]. In [30], a very weak condition is proposed for the non-convex function to converge to the global optimum almost everywhere, and in [31], a new convergence analysis method for SGD under a decreasing learning rate regime is proposed. In [16,32,33], the authors studied several classes of stochastic optimization algorithms enriched with heavy ball momentum, showing a linear rate for the stochastic heavy ball method (i.e., stochastic gradient descent method with momentum (SGDM)). This does not require large memory, merely requiring slightly more computation in each iteration compared with the vanilla SGD method. Therefore, both techniques have been widely used and demonstrated to be effective for training deep neural networks [10,13]. On the one hand, common SGD variants have been designed and analyzed under convex settings [12], and the results may not provide a relevant guarantee of convergence [13]. On the other hand, it is well known that linear convergence can be achieved even with constant step-size gradient descent under certain conditions. However, while most of the advanced SGD variants can achieve faster convergence rates by applying adaptive step size, the convergence rate is not yet ideal.

We summarize the main contributions of the present paper to the existing results in the literature as follows:

- For smooth and convex functions, a novel adaptive step-size stochastic gradient descent (AdaSGD) method is proposed, and a momentum acceleration variant (AdaSGDM) is studied as well. It is proven that both have a convergence rate of $O(1/T)$, where T is the maximum number of iterations.
- For smooth but non-convex functions, we show that both AdaSGD and AdaSGDM achieve global optimization with a convergence rate of $O(1/T)$.

The rest of this paper is organized as follows. In Section 2, we describe the optimization problem and present the AdaSGD and AdaSGDM method along with details of the adaptive step sizes. In Section 3, we prove the convergence rates of the proposed AdaSGD and AdaSGDM theoretically. Section 4 presents a practical implementation and discusses the experimental results on problems arising from machine learning. Finally, a brief conclusion and discussion of possible future work is presented in Section 5.

2. Problem Statement

Consider the following unconstrained minimization problem:

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

where $f : \mathbb{R}^d \to \mathbb{R}$ is a differentiable function (though not necessarily convex). More concretely, we assume that $f(x)$ has a Lipschitz gradient.

Assumption 1. *The continuously differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is bounded below by $f^* := \inf_{x\in\mathbb{R}^d} f(x) \in \mathbb{R}$, and its gradient $\nabla f(x)$ is L-Lipschitz; i.e., there exists a constant $L > 0$ such that, for all $x, y \in \mathbb{R}^d$,*

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|,\ \forall x, y \in \mathbb{R}^d,$$

where $\|\cdot\|$ denotes the Euclidean norm.

Notice that the inequality does not imply the convexity of f. However, the assumption that f is L-smooth for any $x, y \in \mathbb{R}^d$ implies that ([34], Lemma 1.2.3)

$$|f(y) - f(x) - \langle\nabla f(x), y - x\rangle| \leq \frac{L}{2}\|y - x\|^2.$$

Because we are interested in solving (1) using stochastic gradient methods, we assume that at each $x \in \mathbb{R}^d$ we have access to an unbiased estimator of the true gradient $\nabla f(x)$, denoted by $g(x, \xi)$, where ξ is a source of randomness. Thus, we need the following assumptions, which analyze SGD under the assumptions that $f(x)$ is lower bounded and that the stochastic gradients $g(x, \xi)$ are unbiased and have bounded variance [5].

Assumption 2. *For any $k \geq 1$, the stochastic gradient oracle provides us an independent unbiased estimate $g(x_k, \xi_k)$ of $\nabla f(x_k)$ upon receiving query $x_k \in \mathbb{R}^d$:*

$$E[g(x_k, \xi_k)] = \nabla f(x_k),$$

where ξ is a random variable satisfying certain specific distributions and the variance of the random variable is bounded as follows:

$$E[\|g(x_k, \xi_k) - \nabla f(x_k)\|^2] \leq \sigma^2,$$

for some parameter $0 \leq \sigma < \infty$.

It is worth noting that in the standard setting for SGD, the random vectors $\xi, k = 1, 2, \ldots,$ are independent of each other (and of x_k; see, e.g., [17]). Note that due to unbiasedness, Assumption 2 is the standard stochastic gradient oracle assumption used for SGD analysis and the standard variance bound is equivalent to $E\left[\|g(x, \xi)\|^2\right] \leq \|\nabla f(x)\|^2 + \sigma^2$. Classic convergence analysis of the SGD algorithm relies on placing conditions on the positive step size η_k [1]. In particular, sufficient conditions are that

$$\sum_{k=1}^{\infty} \eta_k = \infty, \quad \text{and} \quad \sum_{k=1}^{\infty} \eta_k^2 < \infty.$$

The first condition is both necessary and intuitive, as it is necessary for the algorithm to be able to travel an arbitrary distance in order to reach the stationary point from the initial point. However, the second condition is actually unnecessary. Many popular step size choices, such as that of Adagrad [12], find it difficult to satisfy this condition, even when the step size can guarantee the convergence of Adagrad on convex sets.

2.1. Adaptive Step Size Stochastic Gradient Descent

More specifically, Adagrad [11] can be used to solve problem (1), as follows:

$$x_{k+1} = x_k - \frac{\eta}{\sqrt{G(x_k, \xi_k) + \epsilon}} \odot g(x_k, \xi_k),$$

where the element-wise matrix-vector multiplication $\odot$ between $G(x_k, \xi_k)$ and $g(x_k, \xi_k)$, which here is $G_k \in \mathbb{R}^{d \times d}$, is a diagonal matrix in which each diagonal element $i, i = 1, 2, \ldots, d$ is the sum of the squares of the gradients with respect to x_k up to time step k, while ϵ is a smoothing term that avoids division by zero (usually on the order of 1×10^{-8}). Interestingly, without the square root operation, the algorithm performs much more poorly. In this work, we focus on SGD with adaptive step size promotion, which iteratively updates the solution via

$$x_{k+1} = x_k - \eta_k g(x_k, \xi_k),$$

with an arbitrary initial point x_0 and adaptive step-size η_k; ξ_k is a random variable obeying some distribution. In the sequelae, we let $g_k := g(x_k, \xi_k)$ denote a stochastic gradient and assume that we have access to a stochastic first-order black-box oracle that returns a noisy estimate of the gradient of f at any point $x \in \mathbb{R}^d$. Unlike [11], in this paper we use the expectation of the stochastic gradient g_k and its second moment to design a new adaptive step size, then obtain a new kind of adaptive stochastic gradient descent method (AdaSGD).

The pseudo-code of our proposed AdaSGD algorithm is presented in Algorithm 1.

Algorithm 1 Adaptive Stochastic Gradient Descent (AdaSGD) Method

```
Initialization: initialize $x_0$ and the maximum number of iterations $T$
Iterate:
for $k = 0, 1, 2, \ldots, T$ do
    Compute the step size (i.e., learning rate) $\eta_k > 0$.
    Generate a random variable $\xi_k$.
    Compute a stochastic gradient $g(x_k, \xi_k)$.
    Update the new iterate $x_{k+1} = x_k - \eta_k g(x_k, \xi_k)$.
End for
```

2.2. Adaptive Step Size Stochastic Gradient Descent with Momentum

In addition, we consider the momentum acceleration variant of the proposed AdaSGD for application of the algorithm. Similarly, the difference from stochastic heavy-ball in [35] is the different selection of adaptive step size. The updates to the AdaSGDM are as follows:

$$x_{k+1} = x_k - \eta_k g_k + \beta(x_k - x_{k-1}), \tag{2}$$

with $x_{-1} = x_0$, where $\beta \in [0, 1)$ is the momentum constant. Equivalently, denoting by $y_{k+1} := x_{k+1} - x_k$, AdaSGDM can be implemented in two steps for $k = 1, 2, \ldots$

$$\begin{aligned} y_{k+1} &= \beta y_k - \eta_k g_k, \\ x_{k+1} &= x_k + y_{k+1}, \end{aligned} \tag{3}$$

where $\eta_k > 0$, $\beta \in [0, 1)$. It is notable that during updating of x_{k+1}, a momentum term is constructed based on the auxiliary sequence $\{y_k\}$. When $\beta = 0$, the method returns to AdaSGD. The pseudo-code of the AdaSGDM algorithm is presented in Algorithm 2.

Algorithm 2 Adaptive Stochastic Gradient Descent Momentum (AdaSGDM) Method

```
Initialization: $\beta \neq 0$, initialize $x_{-1}, x_0$ and the maximum number of iterations $T$
Iterate:
for $k = 0, 1, 2, \dots, T$ do
    Compute the step size (i.e., learning rate) $\eta_k > 0$.
    Generate a random variable $\xi_k$.
    Compute a stochastic gradient $g(x_k, \xi_k)$.
    Update the new iterate:
        $y_{k+1} = \beta y_k - \eta_k g_k$,
        $x_{k+1} = x_k + y_{k+1}$.
End for
```

To facilitate analysis of the stochastic momentum methods, we note that (3) implies the following recursions, which are straightforward to verify:

$$x_{k+1} + p_{k+1} = x_k + p_k - \frac{\eta_k}{1-\beta} g_k, \tag{4}$$

where p_k is provided by

$$p_k = \frac{\beta}{1-\beta}(x_k - x_{k-1}), \quad k \geq 1, \tag{5}$$

and $p_0 = 0$. Let $v_k = \frac{1-\beta}{\beta} p_k$; then,

$$v_{k+1} = \beta v_k - \eta_k g_k. \tag{6}$$

3. Convergence Analysis

In this section, without knowledge of the noise, we state the convergence results of AdaSGD and AdaSGDM under the convex settings in Section 3.1. Similarly, the convergence of the two methods under non-convex settings is analyzed in Section 3.2.

3.1. Adaptive Convergence Rates for Convex Functions

In this section, the convergence of AdaSGD and AdaSGDM under convex settings is discussed using the classical convergence analysis method under the specific adaptive step size iteration. Before stating the theorem for the convergence conclusion, we first provide the following technical Lemma for proving the theorem.

Lemma 1 ([15])**.** *When f is L-smooth, then $\|\nabla f(x)\|^2 \leq 2L(f(x) - f(x^*)), \forall x \in R^n$, where $x^* = \arg\min_x f(x)$.*

Next, we first provide the convergence results of AdaSGD and AdaSGDM in the case of convex functions.

Theorem 1. *Assumptions 1 and 2 hold if f is convex by designing an appropriate adaptive step size, as follows:*

$$\eta_k = \delta_k \cdot \frac{\|E[g_k]\|}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|}, \tag{7}$$

where $\delta_k > 0$ is a parameter. Then, the iterates of AdaSGD ($\beta = 0$) and AdaSGDM ($\beta \neq 0$) satisfy the following bound:

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T+1} \frac{1-\beta}{2C} \|x_0 - x^*\|^2, \tag{8}$$

where $\bar{x}_T = \frac{1}{T+1}\sum_{k=1}^{T} x_k$, $x^* = \arg\min_x f(x)$, $x_{-1} = x_0$ *are random initial points, C is a positive constant, and T is the maximum number of iterations.*

Proof. From the iterative format (4), we can obtain

$$\|x_{k+1} + p_{k+1} - x^*\|^2 - \|x_k + p_k - x^*\|^2 = -\frac{2\eta_k}{1-\beta}\langle g_k,\ x_k + p_k - x^*\rangle + \frac{\eta_k^2}{(1-\beta)^2}\|g_k\|^2. \quad (9)$$

The adaptive step-size we analyze here is a generalization of ones widely used in the online and stochastic optimization literature. As such, their good performance has already been validated using numerous empirical results. In particular, we consider in the following parts that the step size satisfies (7). In addition, for $\|E[g_k]\|$ and $\left(E\left[\|g_k\|^2\right]\right)^{1/2}$, there always exists $C_{1k} \in (0,1)$ and $C_{2k} > 1$ such that

$$\frac{\|E[g_k]\|}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|} \geq C_{1k}, \quad (10)$$

and

$$1 < \frac{\left(E\left[\|g_k\|^2\right]\right)^{1/2}}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|} \leq C_{2k}. \quad (11)$$

Taking the conditional expectation with respect to $\xi_1, \ldots, \xi_{k-1}$, we can find that

$$\begin{aligned}
& E\left[\eta_k\langle g_k,\ x_k + p_k - x^*\rangle\right] \\
= \ & \langle E[g_k], x_k + p_k - x^*\|\rangle \cdot \frac{\|E[g_k]\|}{\left(E[\|g_k\|^2]\right)^{1/2} - \|E[g_k]\|} \cdot \delta_k \\
\geq \ & \langle \nabla f(x_k), x_k + p_k - x^*\rangle \cdot C_{1k}\delta_k \\
= \ & \delta_k C_{1k} \cdot \left\langle \nabla f(x_k), x_k - x^*\right\rangle + \delta_k C_{1k} \cdot \frac{\beta}{1-\beta} \cdot \left\langle \nabla f(x_k), x_k - x_{k-1}\right\rangle \\
\geq \ & \delta_k C_{1k}\big(f(x_k) - f(x^*)\big) + \delta_k C_{1k} \cdot \frac{\beta}{1-\beta}\big(f(x_k) - f(x_{k-1})\big) \\
\geq \ & \delta_k C_{1k}\big(f(x_k) - f(x^*)\big) + \bar{\delta} C_0 \cdot \frac{\beta}{1-\beta}\big(f(x_k) - f(x_{k-1})\big). \quad (12)
\end{aligned}$$

The first inequality is provided by (10), and the second inequality by the convexity of the function. The last inequality is provided by defining where $C_0 := \min\limits_{k=0,\ldots,T} C_{1k}$ and $\bar{\delta} := \min\limits_{k=0,\ldots,T} \delta_k$. Hence, by summing (9) over $k = 0$ to T and incorporating (12), we have

$$\begin{aligned}
& \frac{2}{1-\beta}\left[\sum_{k=0}^{T} \delta_k C_{1k}\big(f(x_k) - f(x^*)\big)\right] \\
\leq \ & \frac{-2\beta}{(1-\beta)^2}\bar{\delta} C_0\big(f(x_T) - f(x_{-1})\big) + \frac{1}{(1-\beta)^2} E\left[\sum_{k=0}^{T} \eta_k^2 \|g_k\|^2\right] \\
& + (\|x_0 + p_0 - x^*\|^2 - \|x_{T+1} + p_{T+1} - x^*\|^2).
\end{aligned}$$

Notice the initial conditions $x_{-1} = x_0$, $p_0 = 0$; then,

$$\begin{aligned}
& \frac{2}{1-\beta}\left[\sum_{k=0}^{T} \delta_k C_{1k}\big(f(x_k) - f(x^*)\big)\right] \\
\leq \ & \frac{-2\beta}{(1-\beta)^2}\bar{\delta} C_0\big(f(x_T) - f(x_0)\big) + \frac{1}{(1-\beta)^2} E\left[\sum_{k=0}^{T} \eta_k^2 \|g_k\|^2\right] + \|x_0 - x^*\|^2. \quad (13)
\end{aligned}$$

Next, we consider the boundedness of the second term on the right side of (13):

$$\begin{aligned}
E\left[\sum_{k=0}^{T}\eta_k^2\|g_k\|^2\right] &= \sum_{k=0}^{T}E\left[\eta_k^2\|g_k\|^2\right]\\
&= \sum_{k=0}^{T}E\left[\|g_k\|^2\right]\cdot\delta_k^2\cdot\frac{\|E[g_k]\|^2}{\left(\left(E\left[\|g_k\|^2\right]\right)^{1/2}-\|E[g_k]\|\right)^2}\\
&\leq \sum_{k=0}^{T}C_{2k}^2\cdot\delta_k^2\cdot\|\nabla f(x_k)\|^2\\
&\leq \sum_{k=0}^{T}C_{2k}^2\cdot\delta_k^2\cdot 2L\big(f(x_k)-f(x^*)\big),
\end{aligned}\tag{14}$$

the first inequality is provided by (11) and the second by Lemma 1. Substituting (14) into (13), we have

$$\begin{aligned}
\frac{2}{1-\beta}\sum_{k=0}^{T}\delta_k C_{1k}\big(f(x_k)-f(x^*)\big) \leq\; & \frac{1}{(1-\beta)^2}\sum_{k=0}^{T}C_{2k}^2\cdot\delta_k^2\cdot 2L\big(f(x_k)-f(x^*)\big)\\
& -\frac{2\beta\bar{\delta}C_0}{(1-\beta)^2}\big(f(x_T)-f(x_0)\big)+\|x_0-x^*\|^2.
\end{aligned}\tag{15}$$

By recombining (15) and the definition of C_0,

$$\begin{aligned}
&\frac{2}{(1-\beta)^2}\sum_{k=0}^{T}\big((1-\beta)\delta_k C_{1k}-LC_{2k}^2\delta_k^2\big)\big(f(x_k)-f(x^*)\big)\\
\leq\; & -\frac{2\beta\bar{\delta}C_0}{(1-\beta)^2}\big(f(x_T)-f(x_0)\big)+\|x_0-x^*\|^2\\
\leq\; & \frac{2\beta\bar{\delta}C_0}{(1-\beta)^2}\big(f(x_0)-f(x^{*)}\big)+\|x_0-x^*\|^2,
\end{aligned}$$

and we choose $\delta_k < \frac{1-\beta}{L}\frac{C_{1k}}{C_{2k}^2}$ such that $(1-\beta)\delta_k C_{1k}-LC_{2k}^2\cdot\delta_k^2 > 0$.

Note that $0 < C_{1k} < 1$ and $0 < C_{1k}+1 < C_{2k}$ can be obtained from (10) and (11). Per the definition of C_0 and $\bar{\delta}$, and without loss of generality, we can assume that $\bar{\delta}=\delta_{k_0}$. Then,

$$\bar{\delta}C_0=\delta_{k_0}C_0 \leq \frac{1-\beta}{L}\frac{2C_{1k_0}}{C_{2k_0}^2}C_0 \leq \frac{1-\beta}{L}\frac{2C_{1k_0}^2}{C_{2k_0}^2} \leq \frac{1-\beta}{L}.$$

Let $C := \min_{k=0,\ldots,T}\{(1-\beta)\delta_k C_{1k}-LC_{2k}^2\delta_k^2\}$; then,

$$\frac{2C}{(1-\beta)^2}\sum_{k=0}^{T}\big(f(x_k)-f(x^*)\big) \leq \|x_0-x^*\|^2+\frac{2\beta}{1-\beta}\frac{1}{L}\big(f(x_0)-f(x^{*)}\big),$$

which means that

$$\begin{aligned}
\sum_{k=0}^{T}\big(f(x_k)-f(x^*)\big) &\leq \frac{(1-\beta)^2}{2C}\|x_0-x^*\|^2+\frac{1}{L}\frac{\beta(1-\beta)}{C}\big(f(x_0)-f(x^{*)}\big)\\
&\leq \frac{(1-\beta)^2}{2C}\|x_0-x^*\|^2+\frac{1}{L}\frac{\beta(1-\beta)}{C}\frac{L}{2}\|x_0-x^*\|^2\\
&\leq \frac{1-\beta}{2C}\|x_0-x^*\|^2.
\end{aligned}$$

Now, from Jensen's inequality, we have

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T+1}\sum_{k=0}^{T}\left(f(x_k) - f(x^*)\right) \leq \frac{1}{T+1}\frac{1-\beta}{2C}\|x_0 - x^*\|^2,$$

where $\bar{x}_T = \frac{1}{T+1}\sum_{k=0}^{T} x_k$. □

3.2. Adaptive Convergence for Non-Convex Optimization

We now turn to the case where f is non-convex. In practice, most loss functions are non-convex. Because the convexity of a function plays an important role in convergence analysis, the convergence conclusion is not valid in the case of non-convexity. However, there are few theoretical results about stochastic optimization convergence in non-convex environments. In this section, we analyze the convergence of AdaSGD and AdaSGDM under non-convex settings by applying the expectation of the stochastic gradient and the second moment to the design of the adaptive step size.

Theorem 2. *Let Assumptions 1 and 2 hold if f is non-convex. We choose the step size as in (7). Then, the iterates of AdaSGD satisfy the following bound:*

$$\min_{k=1,\dots,T}\|\nabla f(x_k)\|^2 \leq \frac{1}{T}\cdot\frac{1}{\hat{C}}\left(f(x_0) - f(x^*)\right),$$

where x^* is one of the minimum point of the function $f(x)$ over $\mathbb{R}^d$, x_0 is a random initial point, $\hat{C}$ is a positive constant, and T is the maximum number of iterations.

Proof. Because $f(x)$ is an L-smooth function, we have

$$f(y) - f(x) - \langle\nabla f(x), y - x\rangle \leq \frac{L}{2}\|y - x\|^2.$$

Then,

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \langle\nabla f(x_k), x_{k+1} - x_k\rangle + \frac{L}{2}\|x_{k+1} - x_k\|^2 \\ &= f(x_k) - \langle\nabla f(x_k), \eta_k g_k\rangle + \frac{L}{2}\eta_k^2\|g_k\|^2. \end{aligned} \tag{16}$$

Using the expectation on both sides of (16),

$$\begin{aligned} E[f(x_{k+1}) - f(x_k)] &\leq -E[\langle\nabla f(x_k), \eta_k g_k\rangle] + \frac{L}{2}\eta_k^2 E[\|g_k\|^2] \\ &= -\eta_k\|\nabla f(x_k)\|^2 + \frac{L}{2}\eta_k^2 E\left[\|g_k\|^2\right]. \end{aligned}$$

Now, by taking the adaptive step size as (7), we have

$$\begin{aligned} E[f(x_{k+1}) - f(x_k)] \leq\; & -\delta_k\cdot\frac{\|E[g_k]\|}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|}\cdot\|\nabla f(x_k)\|^2 \\ & + \frac{L}{2}\delta_k^2\cdot\left(\frac{\|E[g_k]\|}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|}\right)^2\cdot E\left[\|g_k\|^2\right] \\ =\; & -\delta_k\cdot\frac{\|E[g_k]\|}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|}\cdot\|\nabla f(x_k)\|^2 \\ & + \frac{L}{2}\delta_k^2\cdot\left(\frac{\left(E[\|g_k\|^2]\right)^{1/2}}{\left(E\left[\|g_k\|^2\right]\right)^{1/2} - \|E[g_k]\|}\right)^2\cdot\|E[g_k]\|^2. \end{aligned}$$

From (10) and (11),

$$E[f(x_{k+1}) - f(x_k)] \leq -\delta_k C_{1k} \|\nabla f(x_k)\|^2 + \frac{L}{2}\delta_k^2 C_{2k}^2 \cdot \left\|E[g_k]\right\|^2,$$

we choose $\delta_k < \frac{2}{L}\frac{C_{1k}}{C_{2k}^2}$ such that $\delta_k C_{1k} - \frac{L}{2}\delta_k^2 C_{2k}^2 > 0$. Let $\hat{C} := \min_{k=0,\ldots,T} \{\delta_k C_{1k} - \delta_k^2 C_{2k}^2\}$; then,

$$f(x_{k+1}) - f(x_k) \leq -\hat{C}\|\nabla f(x_k)\|^2. \tag{17}$$

By summing (17) for $k = 0, \ldots, T$ and averaging it,

$$\frac{\hat{C}}{T}\sum_{k=0}^{T} \|\nabla f(x_k)\|^2 \leq \frac{1}{T+1}(f(x_0) - f(x_{T+1})) \leq \frac{1}{T+1}(f(x_0) - f(x^*)),$$

then,

$$\min_{k=0,\ldots,T} \|\nabla f(x_k)\|^2 \leq \frac{1}{T+1} \cdot \frac{1}{\hat{C}}(f(x_0) - f(x^*)),$$

where x^* is one of the minimum point of the function $f(x)$ over $\mathbb{R}^d$. □

In order to prove the convergence of AdaSGDM under a non-convex setting, we first analyze the relationship between the local error bound of the function and the local variation and gradient. Second, the relationship between the local variation of gradient and gradient is further analyzed. Finally, the boundary of the gradient is obtained, that is, the convergence of AdaSGDM under a non-convex setting. Before we state the adaptive convergence of AdaSGDM for non-convex optimization, we first present the following two Lemmas.

Lemma 2. *Let $z_k = x_k + p_k$. For AdaSGDM, we have the following for any $k \geq 0$:*

$$\begin{aligned} E[f(z_{k+1}) - f(z_k)] \quad \leq \quad & \frac{1}{2L}E\left[\|\nabla f(z_k) - \nabla f(x_k)\|^2\right] \\ & -\left(\frac{1}{1-\beta}C_{1k}\delta_k - \frac{L}{(1-\beta)^2}C_{2k}^2\delta_k^2\right)E\left[\|\nabla f(x_k)\|^2\right], \end{aligned}$$

where L is the Lipschitz constant of f, $\beta \in [0,1)$ is the momentum constant as mentioned in (2), C_{1k} and C_{2k} are parameters in (10) and (11), and δ_k is the parameter in (7).

Proof. Because $f(x)$ is a smooth function, we have

$$f(y) - f(x) - \langle \nabla f(x), y - x\rangle \leq \frac{L}{2}\|y - x\|^2.$$

We define $\omega_k = g_k - \nabla f(x_k)$; then, from Assumption 2, $E[\omega_k] = 0$ can be obtained. Then,

$$\begin{aligned} f(z_{k+1}) \quad \leq \quad & f(z_k) + \langle \nabla f(z_k), z_{k+1} - z_k\rangle + \frac{L}{2}\|z_{k+1} - z_k\|^2 \\ = \quad & f(z_k) - \frac{1}{1-\beta}\eta_k \nabla f(z_k)^T g_k + \frac{L}{2}\frac{1}{(1-\beta)^2}\eta_k^2\|g_k\|^2 \\ = \quad & f(z_k) - \frac{1}{1-\beta}\eta_k \nabla f(z_k)^T(\omega_k + \nabla f(x_k)) + \frac{L}{2}\frac{1}{(1-\beta)^2}\eta_k^2\|g_k\|^2 \\ = \quad & f(z_k) - \frac{\eta_k}{1-\beta}\nabla f(z_k)^T\omega_k - \frac{\eta_k}{1-\beta}\nabla f(x_k)^T(\nabla f(z_k) - \nabla f(x_k)) \\ & -\frac{\eta_k}{1-\beta}\|\nabla f(x_k)\|^2 + \frac{L}{2}\frac{1}{(1-\beta)^2}\eta_k^2\|g_k\|^2. \end{aligned} \tag{18}$$

Recombining (18) and using the expectation of both sides,

$$\begin{aligned} E[f(z_{k+1}) - f(z_k)] &\leq -\frac{1}{1-\beta}\eta_k E[\nabla f(x_k)^T(\nabla f(z_k) - \nabla f(x_k))] - \frac{1}{1-\beta}\eta_k E[\|\nabla f(x_k)\|^2] \\ &\quad + \frac{L}{2}\frac{1}{(1-\beta)^2}\eta_k^2 E[\|g_k\|^2] \\ &\leq \frac{1}{2}E\left[\frac{1}{L}\|\nabla f(z_k) - \nabla f(x_k)\|^2 + \frac{L}{(1-\beta)^2}\eta_k^2\|\nabla f(x_k)\|^2\right] \\ &\quad - \frac{1}{1-\beta}\eta_k E[\|\nabla f(x_k)\|^2] + \frac{L}{2}\frac{1}{(1-\beta)^2}\eta_k^2 E[\|g_k\|^2], \end{aligned} \tag{19}$$

where the second inequality uses the inequality of the arithmetical and geometric means ($ab \leq \frac{1}{2}(a^2 + b^2)$). By taking the adaptive step-size as (7) and substituting it into (19), we have

$$\begin{aligned} E[f(z_{k+1}) - f(z_k)] &\leq \frac{1}{2L}E[\|\nabla f(z_k) - \nabla f(x_k)\|^2] - \frac{1}{1-\beta}C_{1k}\delta_k E[\|\nabla f(x_k)\|^2] \\ &\quad + \frac{L}{(1-\beta)^2}C_{2k}^2\delta_k^2 E[\|\nabla f(x_k)\|^2], \end{aligned}$$

using (10) and (11). □

Lemma 3. *For AdaSGDM, for any $k \geq 1$, we have*

$$E[\|\nabla f(z_k) - \nabla f(x_k)\|^2] \leq \frac{L^2\beta^2}{(1-\beta)^2} \cdot \Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\delta_{k-1-i}^2 C_{1k}^2\|\nabla f(x_{k-1-i})\|^2, \tag{20}$$

where $\Gamma_{k-1} := \sum_{i=0}^{k-1}\beta^i = \frac{1-\beta^k}{1-\beta}$, L is the Lipschitz constant of f, and $\beta \in [0,1)$ is the momentum constant as mentioned in (2). For $k \geq 1$, C_{1k} and C_{2k} are parameters in (10) and (11), respectively, and δ_k is the parameter in (7).

Proof. Because f is L-smooth, $z_k = x_k + p_k$, and (5), we have

$$\|\nabla f(z_k) - \nabla f(x_k)\|^2 \leq L^2\|z_k - x_k\|^2 = L^2\|p_k\|^2 = \frac{L^2\beta^2}{(1-\beta)^2}\|x_k - x_{k-1}\|^2. \tag{21}$$

Recall the recursion in (6), that is, $v_{k+1} = \beta v_k - \eta_k g_k$. Note that $v_0 = 0$. By induction, for $k \geq 1$,

$$v_k = -\sum_{i=0}^{k-1}\beta^i\eta_{k-1-i}g_{k-1-i}. \tag{22}$$

Let $\Gamma_{k-1} = \sum_{i=0}^{k-1}\beta^i = \frac{1-\beta^k}{1-\beta}$; then,

$$\begin{aligned} \|v_k\|^2 &= \left\|\sum_{i=0}^{k-1}\frac{\beta^i}{\Gamma_{k-1}}\eta_{k-1-i}g_{k-1-i}\right\|^2 \cdot \Gamma_{k-1}^2 \\ &\leq \Gamma_{k-1}^2\sum_{i=0}^{k-1}\frac{\beta^i}{\Gamma_{k-1}}\eta_{k-1-i}^2\|g_{k-1-i}\|^2 \\ &= \Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\eta_{k-1-i}^2\|g_{k-1-i}\|^2. \end{aligned} \tag{23}$$

Taking the expectation over both sides of (23) and noting the step size (7), we have

$$\begin{aligned} E[\|v_k\|^2] &\leq \Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\eta_{k-1-i}^2 E[\|g_{k-1-i}\|^2] \\ &\leq \Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\delta_{k-1-i}^2 C_{2,k-1-i}^2\|\nabla f(x_{k-1-i})\|^2. \end{aligned}$$

Then, taking the expectation of both sides of (21) and substituting the above inequality into it, we have

$$\begin{aligned} E[\|\nabla f(z_k)-\nabla f(x_k)\|^2] &\leq \frac{L^2\beta^2}{(1-\beta)^2}E[\|x_k-x_{k-1}\|^2] \\ &= \frac{L^2\beta^2}{(1-\beta)^2}E[\|v_k\|^2] \\ &\leq \frac{L^2\beta^2}{(1-\beta)^2}\Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\delta_{k-1-i}^2 C_{2,k-1-i}^2\|\nabla f(x_{k-1-i})\|^2, \end{aligned}$$

which means that (20) is established. □

Based on the previous Lemmas 2 and 3, we can now state the convergence analysis of AdaSGDM under non-convex settings.

Theorem 3. *Let Assumptions 1 and 2 hold, and let f be a non-convex and L-smooth function. Choosing the step size as in (7), the iteration sequence x_k obtained by AdaSGDM satisfies the following bound:*

$$\min_{k=0,\dots,T-1} E\left[\|\nabla f(x_k)\|^2\right] \leq \frac{1}{\bar{c}-\bar{d}}\frac{1}{T}(f(x_0)-f(x^*)),$$

where $x^* = \arg\min_x f(x)$, $\bar{c}, \bar{d} \geq 0$ are constants, $\bar{c} > \bar{d}$, and T is the maximum number of iterations.

Proof. From the initial conditions, it follows that $z_0 = x_0$; thus, Lemmas 2 and 3 imply the following inequality:

$$\begin{aligned} E[f(z_{k+1})-f(z_k)] \leq\ & \frac{1}{2L}\frac{L^2\beta^2}{(1-\beta)^2}\Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\delta_{k-1-i}^2 C_{2k}^2 E[\|\nabla f(x_{k-1-i})\|^2] \\ & -\left(\frac{1}{1-\beta}C_{1k}\delta_k - \frac{L}{(1-\beta)^2}C_{2k}^2\delta_k^2\right)E[\|\nabla(x_k)\|^2]. \end{aligned} \tag{24}$$

By summing (24) for $k = 0, \dots, T$,

$$\begin{aligned} & E[f(z_{T+1})-f(z_0)] \\ \leq\ & E[\|\nabla f(z_0)-\nabla f(x_0)\|^2] - \sum_{k=0}^{T}\left(\frac{1}{1-\beta}C_{1k}\delta_k - \frac{L}{(1-\beta)^2}C_{2k}^2\delta_k^2\right)E[\|\nabla f(x_k)\|^2] \\ & +\frac{1}{2L}\frac{L^2\beta^2}{(1-\beta)^2}\sum_{k=1}^{T}\Gamma_{k-1}\sum_{i=0}^{k-1}\beta^i\delta_{k-1-i}^2 C_{2k}^2 E[\|\nabla f(x_{k-1-i})\|^2] \\ =\ & -\sum_{k=0}^{T-1}(c_k-d_k)E\left[\|\nabla f(x_k)\|^2\right] - c_T E\left[\|\nabla f(x_T)\|^2\right], \end{aligned}$$

where $c_k := \frac{1}{1-\beta} C_{1k}\delta_k - \frac{L}{(1-\beta)^2} C_{2k}^2 \delta_k^2$ and $d_k := \frac{1}{2} \frac{L\beta^2}{(1-\beta)^2} \delta_k^2 C_{2k}^2 \sum_{i=k}^{T-1} \Gamma_i \beta^{i-k}$. For $k = 0, \dots, T$, we choose $\delta_k < \frac{1-\beta}{L} \frac{C_{1k}}{C_{2k}^2} \frac{1}{2+\beta^2 \sum_{i=k}^{T-1} \Gamma_i \beta^{i-k}}$. Thus, it is true that $c_k > d_k$ for $k = 0, \dots, T-1$ as well as that $c_T > 0$. Then,

$$\begin{aligned} \sum_{k=0}^{T-1} (c_k - d_k) E\left[\|\nabla f(x_k)\|^2\right] &\leq E[f(z_0) - f(z_{T+1})] - c_T E\left[\|\nabla f(x_T)\|^2\right] \\ &\leq f(z_0) - f(z_{T+1}). \end{aligned}$$

Furthermore, because $z_0 = x_0$ and $x^* = \arg\min_x f(x)$, we have

$$\min_{k=0,\dots,T-1} E[\|\nabla f(x_k)\|^2] \leq \frac{1}{\bar{c} - \bar{d}} \frac{1}{T} (f(z_0) - f(z_{T+1})) \leq \frac{1}{\bar{c} - \bar{d}} \frac{1}{T} (f(x_0) - f(x^*)),$$

where $\bar{c} - \bar{d} = \min_{k=0,\dots,T-1}\{c_k - d_k\}$. □

4. Experiments

In this section, we present experimental results of applying our adaptive schemes to several test problems. Section 4.1 focuses on regularized linear regression problem and regularized logistic regression problem, which are widely used in the machine learning community, while Section 4.2 considers the non-convex support vector machine (SVM) problem and non-convex quadratic problem. In both, we report the performance of AdaSGD and AdaSCDM and compare them with SGDM, Adam and Adagrad. In each instance, we set the step size for AdaSGD and AdaSGDM using the procedure in (7). To make the comparison equitable, the default parameter values for Adam are selected according to [9], especially $\eta = 0.001, \beta_1 = 0.9$, $\beta_2 = 0.999$ and $\rho = 10^{-8}$. For Adagrad, the initial step size is $\eta_0 = 0.1$. Using random datasets, we prove that the proposed adaptive SGD can effectively solve practical deep learning problems.

The parameters in SGDM are set to a step size of $\eta = 0.001$ and momentum coefficient of $\beta = 0.8$ in the following applications. We repeated the experiment ten times and report the average results. All methods use the same random initialization; all figures in this section are in log–log scale, and the maximum number of iterations T = 10,000. Finally, all the algorithms involved in the experiment were implemented using MATLAB R2017a (9.2.0.538062) 64 bit software in Windows 10.

4.1. Convex Functions

Consider the following two convex optimization problems: an l_2-regularized quadratic function with $f_1(x) = \|Ax - b\|^2 + \lambda\|x\|_2^2$, and an l_2-regularized logistic regression for binary classification with $f_2(x) = \sum_{i=1}^m \log\left(1 + e^{-b_i A_i x}\right) + \lambda\|x\|_2^2$ with the penalty parameter $\lambda = 0.1$, where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. The entries of b are randomly -1 or 1. Rows A_i in A are generated by an i.i.d multivariate Gaussian distribution conditioned on b_i. We use a mini-batch of size n to compute a stochastic gradient at each iteration. Note that the gradients of functions $f_1(x)$ and $f_2(x)$ are continuous; we assume that random sampling of small batches from the datasets satisfies Assumption 2.

When $m = 60$ and $n = 10$, the convergence paths of the procedure for minimizing different convex functions in SGDM, Adam, Adagrad, and the proposed AdaSGD and AdaSGDM is demonstrated in Figure 1, where the left subfigure in Figure 1, corresponding to function $f_1(x) = \|Ax - b\|^2 + \lambda\|x\|_2^2$, takes 10.337139 s, and the right subfigure in Figure 1, corresponding to function $f_2(x) = \sum_{i=1}^m \log\left(1 + e^{-b_i A_i x}\right) + \lambda\|x\|_2^2$, takes 4.947887 s. When m = 10,000 and $n = 200$. The results are shown in Figure 2, where the left and right subgraphs in Figure 2, corresponding to $f_1(x)$ and $f_2(x)$, take 2130.277402 s and 442.714215 s, respectively.

From the left and right figures in Figures 1 and 2, it is not difficult to see that AdaSGD and AdaSGDM show better convergence than existing stochastic optimization methods when considering the convex optimization problems of different models. Observe that

SGDM displays local acceleration close to the optimal point and attains convergence rate of $O(1/\sqrt{T})$, as shown in [36]. Adagrad shows a convergence rate of $O(1/\sqrt{T})$, as mentioned in [11]. Adam eventually attains a rate of convergence of $O(1/\sqrt{T})$, as shown in [10]. The proposed methods, AdaSGD and AdaSGDM, tend to converge faster than the SGDM, Adam, or Adagrad, showing a convergence of $1/T$, which is consistent with our theory results in this paper.

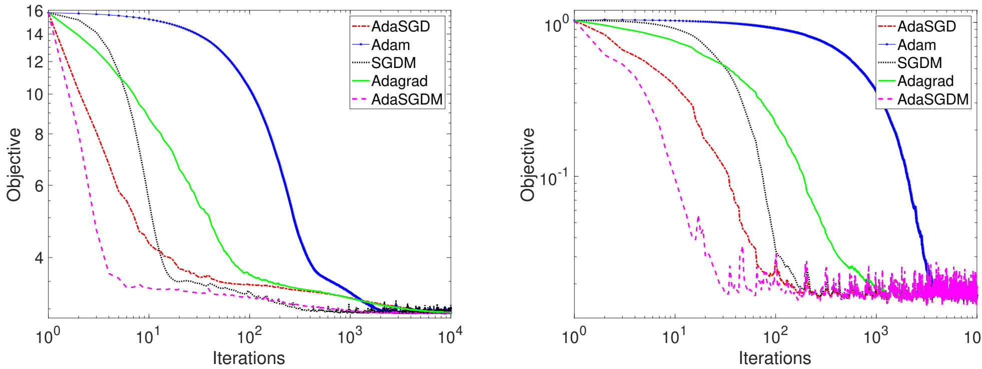

Figure 1. The convergence paths of the algorithms SGDM, Adam, and Adagrad and the proposed methods AdaSGD and AdaSGDM when $m = 60$, $n = 10$, and the objective functions are smooth convex functions. **Left**: the objective function $f_1(x)$; **Right**: the objective function $f_2(x)$.

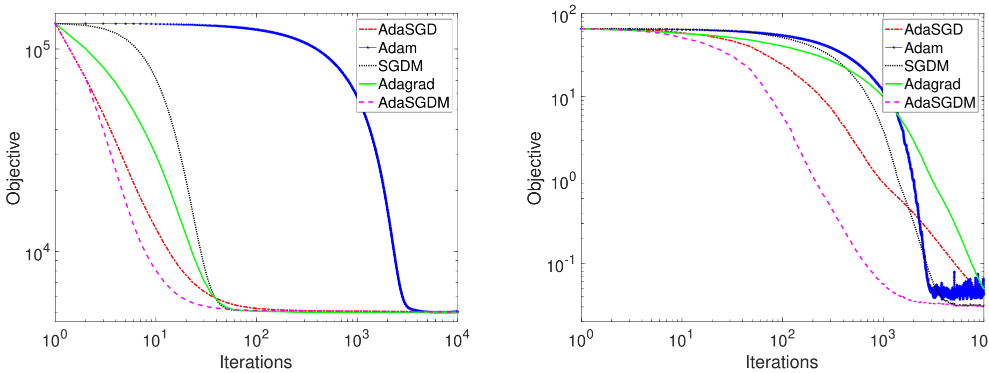

Figure 2. The convergence paths of the algorithms SGDM, Adam, and Adagrad and the proposed methods AdaSGD and AdaSGDM when $m = 10{,}000$, $n = 200$, and the objective functions are smooth convex functions. **Left**: the objective function $f_1(x)$; **Right**: the objective function $f_2(x)$.

4.2. Non-Convex Functions

Consider the following non-convex support vector machine (SVM) problem with a sigmoid loss function, which has previously been considered in [5] (the data points are generated in the same way as in Section 4.1): $\min_{x \in \mathbb{R}^n} f_3(x) := \sum_{i=1}^{m}[1 - \tanh(b_i\langle x, a_i\rangle)] + \lambda\|x\|_2^2$, where $\lambda = 0.1$ is a regularization parameter. In addition, consider the following non-convex optimization problem corresponding to the elastic regression network model [37]: $\min_{x \in \mathbb{R}^n} f_4(x) := \|Ax - b\| - \lambda_1\|x\|_2^2 + \lambda_2\|x\|_1$, where $\lambda_1 = 0.001$ and $\lambda_2 = 0.01$. Here, we use a mini-batch of size n to compute a stochastic gradient at each iteration. For minimizing the two non-convex functions $f_3(x)$ and $f_4(x)$, the gradient of $f_3(x)$ is obviously continuous. For $f_4(x)$ it is easy to know that the derivative of $f_4(x)$ at point $x = 0$ does not exist; however, we can use the subgradient at this point. For example, one of the subgradients of $f_4(x)$ here is $\partial f_4(0) = 0$. Although this gradient is discontinuous, it satisfies the Lipschitz condition, meaning that the conclusion in Theorem 3 holds. The convergence paths of the algorithms SGDM, Adam, Adagrad, AdaSGD, and AdaSGDM when $(m = 60,\ n = 10)$ and $(m = 10{,}000,\ n = 200)$ are shown in Figures 3 and 4. The CPU times of the left and right subfigures corresponding to $f_3(x)$ and $f_4(x)$ in Figure 3 are 10.808409 s and

5.824761 s, respectively, and those of $f_3(x)$ and $f_4(x)$ in Figure 4 are 2369.346180 s and 455.041079 s, respectively.

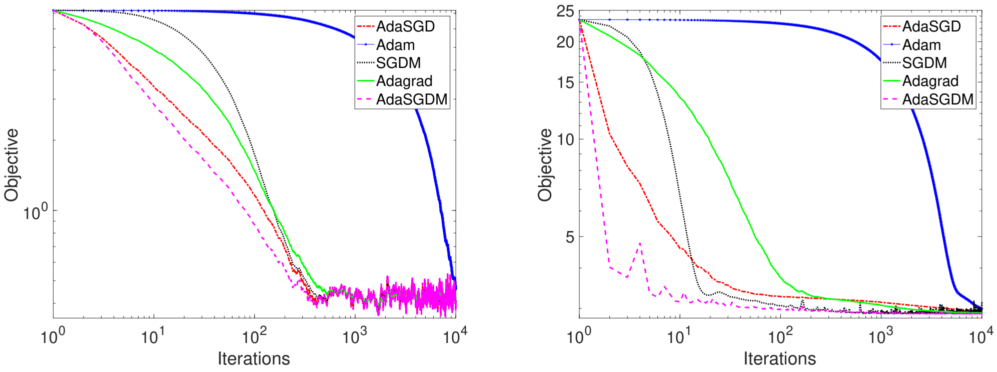

Figure 3. The convergence paths of the algorithms SGDM, Adam, and Adagrad and the proposed methods AdaSGD and AdaSGDM when $m = 60$, $n = 10$, and the objective functions are smooth non-convex functions. **Left**: the objective function $f_3(x)$; **Right**: the objective function $f_4(x)$.

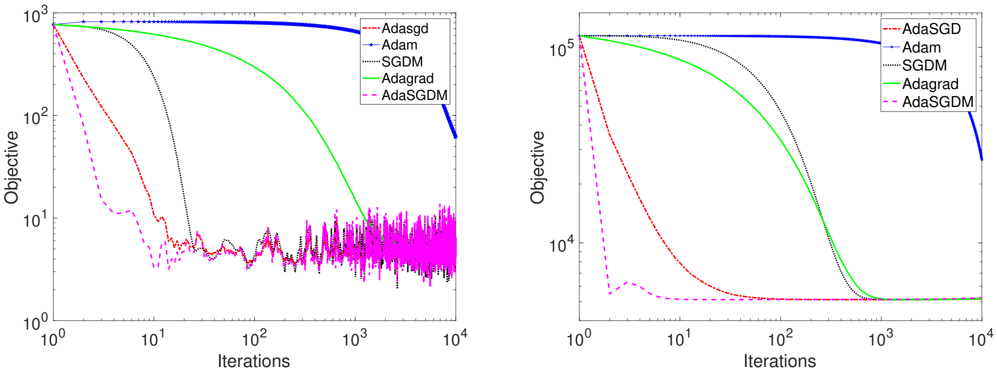

Figure 4. The convergence paths of the algorithms SGDM, Adam, and Adagrad and the proposed methods AdaSGD and AdaSGDM when $m =$ 10,000, $n = 200$, and the objective functions are smooth non-convex functions. **Left**: the objective function $f_3(x)$; **Right**: the objective function $f_4(x)$.

From the figures in Figures 3 and 4, it can be seen that AdaSGD and AdaSGDM maintain good convergence when considering non-convex optimization problems of different models. For different non-convex objective functions with Lipschitz continuous gradients, it can be observed that the gradient converges in expectation at the order of $O(1/\sqrt{T})$ of SGDM, as shown in [36]. As described in [10], the convergence analysis for Adam is not applicable to non-convex problems, and it is only through experience that Adam is likely to perform better than other methods. The Adagrad algorithm displays a convergence rate of $O(\log T/\sqrt{T})$ under non-convex setting, as showed in [38]. For the proposed methods in this paper, AdaSGD and AdaSGDM, tend to converge faster than SGDM, Adam, and Adagrad under non-convex settings, showing a convergence of $1/T$, which is consistent with our theory result in this paper.

5. Conclusions and Future Work

In this paper, two shortcomings of the adaptive stochastic gradient descent method for stochastic optimization problems are studied. The first is the assumption of a convex setting, which is often harsh in many practical optimization problems of machine learning. The second is slow convergence, which is a result of using the adaptive step size of past stochastic gradients, and is generally up to $O(1/\sqrt{T})$. As a consequence, in this paper we first propose a new adaptive SGD in which the new step size is a function of the expectation of the past stochastic gradient and its second moment. In both convex and non-convex settings, the adaptive SGD with the new designed step size converges at the rate of $O(1/T)$.

Second, the new adaptive SGD is extended to the case with momentum, and again achieves a convergence rate of $O(1/T)$, irrespective of convex or non-convex settings. To sum up, our results indicate that the designed adaptive step size is able to alleviate the problem of slow convergence caused by inherent variance to a certain extent. The proposed approach achieves accelerated convergence in convex setting, and works in non-convex settings as well. Experimental results show that the proposed adaptive stochastic gradient descent methods, both with and without momentum, have better convergence performance than existing methods. In the future, we hope to apply this method to large datasets or to actual data collection in order to better analyze its effectiveness.

Author Contributions: Writing—original draft, R.C.; Writing—review & editing, X.T.; Supervision, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Fundamental Research Funds for the Central Universities No. 2662021LXQD001 and the National Natural Science Foundation of China with Grant No. 61903148.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Ye Yuan for his helpful discussions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Robbins, H.; Monro, S. A stochastic approximation method. *Ann. Math. Stat.* **1951**, *22*, 400–407. [CrossRef]
2. Chung, K.L. On a stochastic approximation method. *Ann. Math. Stat.* **1954**, *25*, 463–483. [CrossRef]
3. Polyak, B.T.; Juditsky, A.B. Acceleration of stochastic approximation by averaging. *SIAM J. Control Optim.* **1992**, *30*, 838–855. [CrossRef]
4. Ruszczyński, A.; Syski, W. A method of aggregate stochastic subgradients with on-line stepsize rules for convex stochastic programming problems. *Math. Program. Stud.* **1986**, *28*, 113–131.
5. Ghadimi, S.; Lan, G. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM J. Optim.* **2013**, *23*, 2341–2368. [CrossRef]
6. Bach, F. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *J. Mach. Learn.* **2014**, *15*, 595–627.
7. Xiao, L.; Zhang, T.; A proximal stochastic gradient method with progressive variance reduction. *SIAM J. Optim.* **2014**, *24*, 2057–2075. [CrossRef]
8. Johnson, R.; Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; pp. 315–323.
9. Cutkosky, A.; Busa-Fekete, R. Distributed stochastic optimization via adaptive stochastic gradient descent. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 1910–1919.
10. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
11. Mustapha, A.; Mohamed, L.; Ali, K.; Hamlich, M.; Bellatreche, L.; Mondal, A. An Overview of Gradient Descent Algorithm Optimization in Machine Learning: Application in the Ophthalmology Field. In Proceedings of the Smart Applications and Data Analysis. SADASC 2020, Marrakesh, Morocco, 25–26 June 2020; pp. 349–359.
12. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn.* **2011**, *12*, 257–269.
13. Zeiler, M.D. Adadelta: An adaptive learning rate method. *arXiv* **2012**, arXiv:1212.5701.
14. Reddi, S.J.; Kale, S.; Kumar, S. On the convergence of Adam and beyond. *arXiv* **2019**, arXiv:1904.09237.
15. Li, X.; Orabona, F. On the convergence of stochastic gradient descent with adaptive stepsizes. *arXiv* **2018**, arXiv:1805.08114.
16. Yousefian, F.; Nedi, A.; Shanbhag, U.V. On stochastic gradient and subgradient methods with adaptive steplength sequences. *Automatica* **2012**, *48*, 56–67. [CrossRef]
17. Nemirovski, A.; Juditsky, A.; Lan, G.; Shapiro, A. Robust stochastic approximation approach to stochastic programming. *SIAM J. Optim.* **2013**, *19*, 1574–1609. [CrossRef]
18. Qian, N. On the momentum term in gradient descent learning algorithms. *Neural Netw.* **1999**, *12*, 145–151. [CrossRef] [PubMed]
19. Polyak, B.T. Some methods of speeding up the convergence of iteration methods. *USSR Comput. Math. Math. Phys.* **1964**, *4*, 1–17. [CrossRef]
20. Nesterov, Y.E. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Sov. Math. Dokl.* **1983**, *269*, 543–547.
21. Klein, S.; Pluim, J.P.W.; Staring, M.; Viergever, M.A. Adaptive stochastic gradient descent optimisation for image registration. *Int. J. Comput. Vis.* **2009**, *81*, 227. [CrossRef]

22. Yuan, Y.; Li, M.; Liu, J.; Tomlin, C.J. On the Powerball method for optimization. *arXiv* **2016**, arXiv:1603.07421.
23. Viola, J.; Chen, Y.Q. A Fractional-Order On-Line Self Optimizing Control Framework and a Benchmark Control System Accelerated Using Fractional-Order Stochasticity. *Fractal Fract.* **2022**, *6*, 549. [CrossRef]
24. Holland, J.H. Genetic Algorithms understand Genetic Algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
25. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
26. Xu, K.; Cheng, T.L.; Lope, A.M.; Chen, L.P.; Zhu, X.X.; Wang, M.W. Fuzzy Fractional-Order PD Vibration Control of Uncertain Building Structures. *Fractal Fract.* **2022**, *6*, 473. [CrossRef]
27. Lagunes, M.L.; Castillo, O.; Valdez, F.; Soria, J.; Melin, P. A New Approach for Dynamic Stochastic Fractal Search with Fuzzy Logic for Parameter Adaptation. *Fractal Fract.* **2021**, *5*, 33. [CrossRef]
28. Auer, P.; Cesa-Bianchi, N.; Gentile, C. Adaptive and self-confident on-line learning algorithms. *J. Comput. Syst. Sci.* **2002**, *64*, 48–75. [CrossRef]
29. Prangprakhon, M.; Feesantia, T.; Nimana, N. An Adaptive Projection Gradient Method for Solving Nonlinear Fractional Programming. *Fractal Fract.* **2022**, *6*, 566. [CrossRef]
30. Bottou, L. Online learning and stochastic approximations. *Online Learn. Neural Netw.* **1998**, *17*, 142.
31. Nguyen, L.M.; Nguyen, P.H.; Richtárik, P.; Scheinberg, K.; Takáč, M.; van Dijk, M. New convergence aspects of stochastic gradient algorithms. *J. Mach. Learn. Res.* **2019**, *20*, 1–49.
32. Yan, Y.; Yang, T.; Li, Z.; Lin, Q.; Yang, Y. A unified analysis of stochastic momentum methods for deep learning. *arXiv* **2018**, arXiv:1808.10396.
33. Xu, P.; Wang, T.; Gu, Q. Continuous and discrete-time accelerated stochastic mirror descent for strongly convex functions. In Proceedings of the International Conference on Machine Learning, Macau, China, 26–28 February 2018; pp. 5488–5497.
34. Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*; Springer Science & Business Media: Berlin, Germany, 2013; Volume 87.
35. Zou, F.; Shen, L. On the convergence of adagrad with momentum for training deep neural networks. *arXiv* **2018**, arXiv:1808.03408.
36. Yang, T.; Lin, Q.; Li, Z. Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization. *arXiv* **2016**, arXiv:1604.03257.
37. Facchinei, F.; Scutari, G.; Sagratella, S. Parallel selective algorithms for nonconvex big data optimization. *IEEE Trans. Signal Process.* **2015**, *63*, 1874–1889.
38. Ward, R.; Wu, X.; Bottou, L. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 6677–6686. [CrossRef]