

A new regret analysis for Adam-type algorithms

Ahmet Alacaoglu, Yura Malitsky, Panayotis Mertikopoulos, Volkan Cevher

► **To cite this version:**

Ahmet Alacaoglu, Yura Malitsky, Panayotis Mertikopoulos, Volkan Cevher. A new regret analysis for Adam-type algorithms. ICML '20: The 37th International Conference on Machine Learning, 2020, Vienna, Austria. hal-03043697

HAL Id: hal-03043697

<https://hal.inria.fr/hal-03043697>

Submitted on 7 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A new regret analysis for Adam-type algorithms

Ahmet Alacaoglu¹ Yura Malitsky¹ Panayotis Mertikopoulos^{2,3} Volkan Cevher¹

Abstract

In this paper, we focus on a theory-practice gap for Adam and its variants (AMSgrad, AdamNC, etc.). In practice, these algorithms are used with a constant first-order moment parameter β_1 (typically between 0.9 and 0.99). In theory, regret guarantees for online convex optimization require a rapidly decaying $\beta_1 \rightarrow 0$ schedule. We show that this is an artifact of the standard analysis, and we propose a novel framework that allows us to derive optimal, data-dependent regret bounds with a constant β_1 , without further assumptions. We also demonstrate the flexibility of our analysis on a wide range of different algorithms and settings.

1. Introduction

One of the most popular optimization algorithms for training neural networks is ADAM (Kingma & Ba, 2014), which is a variant of the general class of ADAGRAD-type algorithms (Duchi et al., 2011). The main novelty of ADAM is to apply an exponential moving average (EMA) to gradient estimate (first-order) and to element-wise square-of-gradients (second-order), with parameters β_1 and β_2 , respectively.

In practice, constant β_1 and β_2 values are used (the default parameters in PYTORCH and TENSORFLOW, for example, are $\beta_1 = 0.9$ and $\beta_2 = 0.999$). However, the regret analysis in Kingma & Ba (2014) requires $\beta_1 \rightarrow 0$ with a linear rate, causing a clear discrepancy between theory and practice.

Recently, Reddi et al. (2018) showed that the analysis of ADAM contains a technical issue. Following this discovery, many variants of ADAM are proposed with regret guarantees (Reddi et al., 2018; Chen & Gu, 2018; Huang et al., 2019). Unfortunately, in all these analyses, the requirement $\beta_1 \rightarrow 0$ is inherited and needed to derive the optimal $\mathcal{O}(\sqrt{T})$ regret. In contrast, for favorable practical performance, methods continue to use constant β_1 in experiments.

¹EPFL, Switzerland ²Univ. Grenoble Alpes, CNRS, Inria, LIG, France ³Criteo AI Lab, France. Correspondence to: Ahmet Alacaoglu <ahmet.alacaoglu@epfl.ch>.

One can wonder whether there is an inherent obstacle – in the proposed methods or the setting – which prohibits optimal regret bounds with a constant β_1 ?

In this work, we show that this specific discrepancy between the theory and practice is indeed an artifact of the previous analyses. We point out the shortcomings responsible for this artifact, and then introduce a new analysis framework that attains optimal regret bounds with constant β_1 at no additional cost (and even comes with better constants in the obtained bounds).

Our contributions. In the convex setting, our technique obtains data-dependent $\mathcal{O}(\sqrt{T})$ regret bounds for AMSGRAD and ADAMNC (Reddi et al., 2018). Moreover, our technique can also be applied to a strongly convex variant of ADAMNC, known as SADAM (Wang et al., 2020), yielding again data-dependent logarithmic regret with constant β_1 . To the best of our knowledge, these are the first optimal regret bounds with constant β_1 .

Finally, we illustrate the flexibility of our framework by applying it to zeroth-order (bandit) and nonconvex optimization. In the zeroth-order optimization setting, we improve on the current best result which requires $\beta_1 \sim \frac{1}{t}$, and show that a constant β_1 again suffices. In the non-convex setting, we recover the existing results in the literature, with a simpler proof and slight improvements in the bounds.

It is worth noting that even though our analysis is more flexible and it provides better bounds than prior works, it is not sufficient to explain why nonzero β_1 helps in practice. This is an interesting question requiring further investigation and is outside the scope of this paper.

1.1. Problem Setup

In online optimization, a loss function $f_t: \mathcal{X} \rightarrow \mathbb{R}^d$ is revealed, after a decision vector $x_t \in \mathcal{X}$ is picked by the algorithm. We then minimize the regret defined as

$$R(T) = \sum_{t=1}^T f_t(x_t) - \min_{x \in \mathcal{X}} \sum_{t=1}^T f_t(x). \quad (1)$$

Our assumptions are summarized below which are the same as in (Reddi et al., 2018).

Assumption 1.

- ▷ $\mathcal{X} \subset \mathbb{R}^d$ is a compact convex set.
- ▷ $f_t: \mathcal{X} \rightarrow \mathbb{R}$ is a convex lsc function, $g_t \in \partial f_t(x_t)$.
- ▷ $D = \max_{x,y \in \mathcal{X}} \|x - y\|_\infty$, $G = \max_t \|g_t\|_\infty$.

1.2. Preliminaries

We work in Euclidean space \mathbb{R}^d with inner product $\langle \cdot, \cdot \rangle$. For vectors $a, b \in \mathbb{R}^d$ all standard operations ab , a^2 , a/b , $a^{1/2}$, $1/a$, $\max\{a, b\}$ are supposed to be coordinate-wise. For a given vector $a_t \in \mathbb{R}^d$, we denote its i^{th} coordinate by $a_{t,i}$. We denote the vector of all-ones as $\mathbf{1}$. We use $\text{diag}(a)$ to denote a $d \times d$ matrix which has a in its diagonal, and the rest of its elements are 0. For $v_i > 0, \forall i = 1, \dots, d$, we define a weighted norm

$$\|x\|_v^2 := \langle x, (\text{diag } v)x \rangle$$

and a weighted projection operator onto \mathcal{X}

$$P_{\mathcal{X}}^v(x) = \arg \min_{y \in \mathcal{X}} \|y - x\|_v^2. \quad (2)$$

We note that $\forall x, y \in \mathbb{R}^d$, $P_{\mathcal{X}}^v$ is nonexpansive, that is

$$\|P_{\mathcal{X}}^v(y) - P_{\mathcal{X}}^v(x)\|_v \leq \|y - x\|_v. \quad (3)$$

2. Related work

2.1. Convex world

In the setting of online convex optimization (OCO), Assumption 1 is standard (Hazan et al., 2016; Duchi et al., 2011). It allows us to consider nonsmooth stochastic minimization (though we are not limited to this setting), and even allows for adversarial loss functions.

The algorithms AMSGRAD and ADAMNC were proposed by Reddi et al. (2018) to fix the issue in the original proof of ADAM (Kingma & Ba, 2014). However, as the proof template of Reddi et al. (2018) follows very closely the proof of Kingma & Ba (2014), the requirement for $\beta_1 \rightarrow 0$ remains in all the regret guarantees of these algorithms. In particular, as noted by Reddi et al. (2018, Corollary 1, 2), a schedule of $\beta_{1t} = \beta_1 \lambda^{t-1}$ is needed for obtaining optimal regret. Reddi et al. (2018) also noted that regret bounds of the same order can be obtained by setting $\beta_{1t} = \beta_1/t$. On the other hand, in the numerical experiments, a constant value $\beta_{1t} = \beta_1$ is used consistent with the huge literature following Kingma & Ba (2014).

Following Reddi et al. (2018), there has been a surge of interest in proposing new variants of ADAM with good practical properties; to name a few, PADAM by Chen & Gu (2018), ADABOUND and AMSBOUND by Luo et al. (2019); Savarese (2019), NOSTALGIC ADAM by Huang et al. (2019). As the

regret analyses of these methods follow very closely the analysis of Reddi et al. (2018), the resulting bounds inherited the same shortcomings. In particular, in all these algorithms, to achieve $\mathcal{O}(\sqrt{T})$ regret, one needs either $\beta_{1t} = \beta_1 \lambda^{t-1}$ or $\beta_{1t} = \frac{\beta_1}{t}$. On the other hand, the experimental results reported on these algorithms note that a constant value of β_1 is used in practice in order to obtain better performance.

Similar issues are present in other problem settings. For strongly convex optimization, Wang et al. (2020) proposed the SADAM algorithm as a variant of ADAMNC, which exploits strong convexity to obtain $\mathcal{O}(\log T)$ regret. SADAM was shown to exhibit favorable practical performance in the experimental results of Wang et al. (2020). However, the same discrepancy exists as with previous ADAM variants: a linearly decreasing β_{1t} schedule is required in theory but a constant $\beta_{1t} = \beta_1$ is used in practice.

One work that tried to address this issue is that of Fang & Klabjan (2019), where the authors focused on OCO with strongly convex loss functions and derived an $\mathcal{O}(\sqrt{T})$ regret bound with a constant value of $\beta_1 \leq \frac{\mu\alpha}{1+\mu\alpha}$, where μ is the strong convexity constant and α is the step size that is set as α_1/\sqrt{T} . (Fang & Klabjan, 2019, Theorem 2). However, this result is still not satisfactory, since the obtained bound for β_1 is weak: both strong convexity μ and the step size $\frac{\alpha_1}{\sqrt{T}}$ are small. This does not allow for the standard choices of $\beta_1 \in (0.9, 0.99)$.

Moreover, a quick look into the proof of Fang & Klabjan (2019, Theorem 2) reveals that the proof in fact follows the same lines as Reddi et al. (2018) with the difference of using the contribution of strong convexity to get rid of the spurious terms that require $\beta_1 \rightarrow 0$. Therefore, it is not surprising that the theoretical bound for β_1 depends on μ and α and can only take values close to 0. Second, in addition to the standard Assumption 1, Fang & Klabjan (2019) also assumes strong convexity, which is a quite stringent assumption by itself. In contrast, our approach does not follow the lines of Reddi et al. (2018), but is an alternative way that does not encounter the same roadblocks.

2.2. Nonconvex world

A related direction to what we have reviewed in the previous subsection is to analyze ADAM-type algorithms without convexity assumptions. When convexity is removed, the standard setting in which the algorithms are analyzed, is stochastic optimization with a smooth loss function and no constraints (Chen et al., 2019a; Zhou et al., 2018; Zou et al., 2019). As a result, these algorithms, compared to the convex counterparts, do not perform projections in the update step of x_{t+1} (cf., Algorithm 1).

In addition to smoothness, bounded gradients are assumed, which is also restrictive, as many nonconvex functions do

not satisfy this property. Indeed, one can show that it is equivalent to the Lipschitz continuity of the function (not its gradient!). Under these assumptions, the standard results bound the minimum gradient norm across all iterations.

An interesting phenomenon in this line of work is that a constant $\beta_1 < 1$ is permitted for the theoretical results, which may seem like weakening our claims. However, it is worth noting that these results do not imply any guarantee for regret in OCO setting.

Indeed, adding the convexity assumption to the setting of unconstrained, smooth stochastic optimization, would only help obtaining a gradient norm bound in the averaged iterate, rather than the minimum across all iterations. However, this bound does not imply any guarantee in the objective value, unless more stringent Polyak-Lojasiewicz or strong convexity requirements are added in the mix.

Moreover, in the OCO setting that we analyze, loss functions are nonsmooth, and there exists a constraint onto which a projection is performed in the x_{t+1} step (*cf.*, Algorithm 1). Finally, online optimization includes stochastic optimization as a special case. Given the difference of assumptions, the analyses in (Chen et al., 2019a; Zhou et al., 2018; Zou et al., 2019) indeed do not help obtaining any regret guarantee for standard OCO.

A good example demonstrating this difference on the set of assumptions is the work (Chen et al., 2019b). In this paper, a variant of AMSGRAD is proposed for zeroth order optimization and it is analyzed in the convex and nonconvex settings. Consistent with the previous literature in both, convergence result for the nonconvex setting allows a constant $\beta_1 < 1$ (Chen et al., 2019b, Theorem 1). However, the result in the convex setting requires a decreasing schedule such that $\beta_{1t} = \frac{\beta_1}{t}$ (Chen et al., 2019b, Proposition 4).

As we highlighted above, the analyses in convex/nonconvex settings follow different paths and the results or techniques are not transferrable to each other. Thus, our main aim in this paper is to bridge the gap in the understanding of regret analysis for OCO and propose a new analytic framework. As we see in the sequel, our analysis not only gives the first results in OCO setting, it is also general enough to apply to the abovementioned nonconvex optimization case and recover similar results as the existing ones.

3. Main results

3.1. Dissection of the standard analysis

We start by describing the shortcoming of the previous approaches in (Reddi et al., 2018; Wang et al., 2020) and, then explain the mechanism that allows us to obtain regret bounds with constant β_1 . In this subsection, for full generality, we assume that the update for m_t is not done with β_1 , but with β_{1t} , as in (Reddi et al., 2018; Kingma & Ba, 2014):

$$m_t = \beta_{1t}m_{t-1} + (1 - \beta_{1t})g_t. \quad (4)$$

The standard way to analyze Adam-type algorithms is to start by the nonexpansiveness property (3) and to write

$$\begin{aligned} \|x_{t+1} - x\|_{\hat{v}_t^{1/2}}^2 &\leq \|x_t - x\|_{\hat{v}_t^{1/2}}^2 - 2\alpha_t \langle m_t, x_t - x \rangle \\ &\quad + \alpha_t^2 \|m_t\|_{\hat{v}_t^{-1/2}}^2. \end{aligned}$$

Then using (4), one can deduce

$$\begin{aligned} (1 - \beta_{1t}) \langle g_t, x_t - x \rangle &\leq -\beta_{1t} \langle m_{t-1}, x_t - x \rangle \\ + \frac{\alpha_t}{2} \|m_t\|_{\hat{v}_t^{-1/2}}^2 + \frac{1}{2\alpha_t} &\left(\|x_t - x\|_{\hat{v}_t^{1/2}}^2 - \|x_{t+1} - x\|_{\hat{v}_t^{1/2}}^2 \right). \end{aligned}$$

Let us analyze the above inequality. Its left-hand side is exactly what we want to bound, since by convexity $R(T) \leq \sum_{t=1}^T \langle g_t, x_t - x \rangle$. The last two terms in the right-hand side are easy to analyze, all of them can be bounded in a standard way using just definitions of \hat{v}_t , m_t , and α_t .

What can we do with the term $-\beta_{1t} \langle m_{t-1}, x_t - x \rangle$? Analysis in (Reddi et al., 2018) bounds it with Young's inequality

$$\begin{aligned} -\beta_{1t} \langle m_{t-1}, x_t - x \rangle &\leq \frac{\beta_{1t}}{2\alpha_t} \|x_t - x\|_{\hat{v}_t^{1/2}}^2 \\ &\quad + \frac{\beta_{1t}\alpha_t}{2} \|m_{t-1}\|_{\hat{v}_t^{-1/2}}^2. \end{aligned}$$

The term $\frac{\beta_{1t}}{2\alpha_t} \|x_t - x\|_{\hat{v}_t^{1/2}}^2$ is precisely what leads to the second term in the regret bound in (Reddi et al., 2018, Theorem 4). Since $\alpha_t = \frac{\alpha}{\sqrt{t}}$, one must require $\beta_{1t} \rightarrow 0$.

Note that the update for x_{t+1} has a projection. This is important, since otherwise a solution must lie in the interior of \mathcal{X} , which is not the case in general for problems with a compact domain. However, let us assume for a moment that the update for x_{t+1} does not have any projection. In this simplified setting, applying the following trick will work.

Recall that $x_t = x_{t-1} - \alpha_{t-1} \hat{v}_{t-1}^{-1/2} m_{t-1}$, or equivalently $m_{t-1} = \frac{1}{\alpha_{t-1}} \hat{v}_{t-1}^{1/2} (x_{t-1} - x_t)$. Plugging it into the error term $\langle m_{t-1}, x_t - x \rangle$ yields

$$\begin{aligned} -\langle m_{t-1}, x_t - x \rangle &= -\frac{1}{\alpha_{t-1}} \langle \hat{v}_{t-1}^{1/2} (x_{t-1} - x_t), x_t - x \rangle \\ &= \frac{1}{2\alpha_{t-1}} \left[\|x_t - x_{t-1}\|_{\hat{v}_{t-1}^{1/2}}^2 + \|x_t - x\|_{\hat{v}_{t-1}^{1/2}}^2 \right. \\ &\quad \left. - \|x_{t-1} - x\|_{\hat{v}_{t-1}^{1/2}}^2 \right] \\ &\leq \frac{1}{2} \alpha_{t-1} \|m_{t-1}\|_{\hat{v}_{t-1}^{-1/2}}^2 + \frac{1}{2} \|x_t - x\|_{\hat{v}_{t-1}^{1/2}/\alpha_t}^2 \\ &\quad - \frac{1}{2} \|x_{t-1} - x\|_{\hat{v}_{t-1}^{1/2}/\alpha_{t-1}}^2, \end{aligned}$$

where the second equality follows from the Cosine Law and the first inequality is from $x_t - x_{t-1} = -\alpha_{t-1}\hat{v}_{t-1}^{-1/2}m_{t-1}$ and $\hat{v}_t^{1/2}/\alpha_t \geq \hat{v}_{t-1}^{1/2}/\alpha_{t-1}$. We now compare this bound with the previous one. The term $\alpha_{t-1}\|m_{t-1}\|_{\hat{v}_{t-1}^{-1/2}}^2$, as we already observed, is good for summation. And other two terms are going to cancel after summation over t . Hence, it is easy to finish the analysis to conclude $\mathcal{O}(\sqrt{T})$ regret with a fixed $\beta_{1t} = \beta_1$.

Unfortunately, the update for x_{t+1} does have a projection, without it the assumption for the domain to be bounded is very restrictive. This prevents us from using the above trick. Its message, however, is that it is feasible to expect a good bound even with a fixed β_{1t} , and under the same assumptions on the problem setting.

For having a more general technique to handle β_1 , we will take a different route in the very beginning — we will analyze the term $\langle g_t, x_t - x \rangle$ in a completely different way, without resorting to any crude inequality as in (Reddi et al., 2018). Basically, this idea can be applied to any framework with a similar update for the moment m_t .

3.2. A key lemma

As we understood above, the presence of the projection complicates handling $\langle m_{t-1}, x_t - x \rangle$. A high level explanation for the cause of the issue is that the standard analysis does not leave much flexibility, since it uses nonexpansiveness in the very beginning.

Lemma 1. *Under the definition*

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t,$$

it follows that

$$\begin{aligned} \langle g_t, x_t - x \rangle &= \langle m_{t-1}, x_{t-1} - x \rangle \\ &\quad - \frac{\beta_1}{1 - \beta_1} \langle m_{t-1}, x_t - x_{t-1} \rangle \\ &\quad + \frac{1}{1 - \beta_1} (\langle m_t, x_t - x \rangle - \langle m_{t-1}, x_{t-1} - x \rangle). \end{aligned}$$

The main message of Lemma 1 is that the decomposition of m_t , in the second part of the analysis in Section 3.1 is now done before using nonexpansiveness, therefore there would be no need for using Young's inequality which is the main shortcoming of the previous analysis.

Upon inspection on the bound, it is now easy to see that the last two terms will telescope. The second term can be shown to be of the order $\alpha_t\|m_t\|_{\hat{v}_t^{-1/2}}^2$, and as we have seen before, summing this term will give $\mathcal{O}(\sqrt{T})$. To see that the first term is also benign, a high level explanation is to notice that m_{t-1} is the gradient estimate used in the update

$x_t = x_{t-1} - \alpha_{t-1}\hat{v}_{t-1}^{-1/2}m_{t-1}$, therefore it can be analyzed in the classical way.

We proceed to illustrate the flexibility of the new analysis on three popular ADAM variants that are proven to converge.

3.3. AMSGRAD

AMSGRAD is proposed by (Reddi et al., 2018) as a fix to ADAM. The algorithm incorporates an extra step to enforce monotonicity of second moment estimator \hat{v}_t .

Algorithm 1 AMSGRAD (Reddi et al., 2018)

-
- 1: **Input:** $x_1 \in \mathcal{X}$, $\alpha_t = \frac{\alpha}{\sqrt{t}}$, $\alpha > 0$, $\beta_1 < 1$, $\beta_2 < 1$,
 $m_0 = v_0 = 0$, $\hat{v}_0 = \varepsilon \mathbf{1}$, $\varepsilon \geq 0$
 - 2: **for** $t = 1, 2 \dots$ **do**
 - 3: $g_t \in \partial f_t(x_t)$
 - 4: $m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t$
 - 5: $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2$
 - 6: $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$
 - 7: $x_{t+1} = P_{\mathcal{X}}^{\hat{v}_t^{1/2}}(x_t - \alpha_t \hat{v}_t^{-1/2} m_t)$
 - 8: **end for**
-

The regret bound for this algorithm in (Reddi et al., 2018, Theorem 4, Corollary 1) requires a decreasing β_1 at least at the order of $1/t$ to obtain $\mathcal{O}(\sqrt{T})$ worst case regret. Moreover, it is easy to see that a constant β_1 results in $\mathcal{O}(T\sqrt{T})$ worst case regret in (Reddi et al., 2018, Theorem 4).

We now present the following theorem which shows that the same $\mathcal{O}(\sqrt{T})$ can be obtained by AMSGRAD under the same structural assumptions as (Reddi et al., 2018).

Theorem 1. *Under Assumption 1, $\beta_1 < 1$, $\beta_2 < 1$, $\gamma = \frac{\beta_1^2}{\beta_2} < 1$, and $\varepsilon > 0$, AMSGRAD achieves the regret*

$$\begin{aligned} R(T) &\leq \frac{D^2\sqrt{T}}{2\alpha(1 - \beta_1)} \sum_{i=1}^d \hat{v}_{T,i}^{1/2} \\ &\quad + \frac{\alpha\sqrt{1 + \log T}}{\sqrt{(1 - \beta_2)(1 - \gamma)}} \sum_{i=1}^d \sqrt{\sum_{t=1}^T g_{t,i}^2}. \end{aligned} \quad (5)$$

We would like to note that our bound for $R(T)$ is also better than the one in (Reddi et al., 2018) in term of constants. We have only two terms in contrast to three in (Reddi et al., 2018) and each of them is strictly smaller than their counterparts in (Reddi et al., 2018). The reason is that we used i) new way of decomposition $\langle g_t, x_t - x \rangle$ as in Lemma 1, ii) wider admissible range for β_1, β_2 , iii) more refined estimates for analyzing terms. For example, the standard analysis to estimate $\|m_t\|_{\hat{v}_t^{-1/2}}^2$ uses several Cauchy-Schwarz inequalities. We instead give a better bound by applying generalized Hölder inequality (Beckenbach & Bellman, 1961).

Another observation is that having a constant β_1 explicitly improves the last term in the regret bound. If one uses a non-decreasing β_1 , instead of constant β_1 , then this term will have an additional multiple of $\frac{1}{(1-\beta_1)^2}$. Given that in general one chooses β_1 close to 1, this factor is significant.

Remark 1. Notice that Theorem 1 requires $\varepsilon > 0$ in order to have the weighted projection operator in (2) well-defined. Such a requirement is common in the literature for theoretical analysis, see (Duchi et al., 2011, Theorem 5). In practice, however, one can set $\varepsilon = 0$.

Proof sketch. We sum $\langle g_t, x_t - x \rangle$ from Lemma 1 over t , use $m_0 = 0$ to get

$$\begin{aligned} \sum_{t=1}^T \langle g_t, x_t - x \rangle &\leq \underbrace{\sum_{t=1}^T \langle m_t, x_t - x \rangle}_{S_1} \\ &+ \frac{\beta_1}{1-\beta_1} \underbrace{\sum_{t=1}^T \langle m_{t-1}, x_{t-1} - x_t \rangle}_{S_2} \\ &+ \frac{\beta_1}{1-\beta_1} \underbrace{\langle m_T, x_T - x \rangle}_{S_3}. \end{aligned}$$

By Hölder inequality, we can show that

$$S_2 \leq \sum_{t=1}^{T-1} \alpha_t \|m_t\|_{\hat{v}_t}^2.$$

By using the fact that $\hat{v}_{t,i} \geq \hat{v}_{t-1,i}$, and the same estimation as deriving S_2 ,

$$S_1 \leq \frac{D^2}{2\alpha_T} \sum_{i=1}^d \hat{v}_{T,i}^{1/2} + \sum_{t=1}^T \frac{\alpha_t}{2} \|m_t\|_{\hat{v}_t}^2.$$

By Hölder and Young's inequalities, we can bound S_3 as

$$S_3 \leq \alpha_T \|m_T\|_{\hat{v}_T}^2 + \frac{D^2}{4\alpha_T} \sum_{i=1}^d \hat{v}_{T,i}^{1/2}.$$

Lastly, we see that $\alpha_t \|m_t\|_{\hat{v}_t}^2$ is common in all these terms and it is well known that this term is good for summation

$$\sum_{t=1}^T \alpha_t \|m_t\|_{\hat{v}_t}^2 \leq \frac{(1-\beta_1)\alpha\sqrt{1+\log T}}{\sqrt{(1-\beta_2)(1-\gamma)}} \sum_{i=1}^d \left(\sum_{t=1}^T g_{t,i}^2 \right)^{\frac{1}{2}}.$$

Combining the terms gives the final bound. \square

Finally, if we are interested in the worst case scenario, it is clear that Theorem 1 gives regret $R(T) = \mathcal{O}(\sqrt{\log(T)T})$.

A quick look into the calculations yields that if one uses the worst case bound $g_{t,i} \leq G$, then the bound will not include a logarithmic term. However, then the data-dependence of the bound will be lost. It is not clear if one can obtain a data-dependent $\mathcal{O}(\sqrt{T})$ regret bound. In the following corollary, we give a partial answer to this question.

Corollary 1. Under Assumption 1, $\beta_1 < 1$, $\beta_2 < 1$, $\gamma = \frac{\beta_1^2}{\beta_2} < 1$, and $\varepsilon > 0$, AMSGRAD achieves the regret

$$\begin{aligned} R(T) &\leq \frac{D^2\sqrt{T}}{2\alpha(1-\beta_1)} \sum_{i=1}^d \hat{v}_{T,i}^{1/2} \\ &+ \frac{\alpha\sqrt{G}}{\sqrt{1-\beta_2}(1-\gamma)} \sum_{i=1}^d \sqrt{\sum_{t=1}^T |g_{t,i}|}. \end{aligned} \quad (6)$$

We remark that even though this bound does not contain a $\log(T)$ term, thus better in the worst-case, its data-dependence is actually worse than the standard bound. Standard bound contains $g_{t,i}^2$ whereas bound above contains $|g_{t,i}|$. Therefore, when the values $g_{t,i}$ are very small, the bound with $\log T$ can be better. We leave it as an open question to have a \sqrt{T} bound with the same data-dependence as the original bound.

3.4. ADAMNC

Another variant that is proposed by Reddi et al. (2018) as a fix to ADAM is ADAMNC which features an increasing schedule for β_{2t} . In particular, one sets $\beta_{2t} = 1 - \frac{1}{t}$ in

$$v_t = \beta_{2t}v_{t-1} + (1-\beta_{2t})g_t^2,$$

that results in the following expression for v_t

$$v_t = \frac{1}{t} \sum_{j=1}^t g_j^2,$$

which is a reminiscent of ADAGRAD (Duchi et al., 2011).

In fact, to ensure that $P_{\mathcal{X}}^{v_t^{1/2}}$ is well-defined, one needs to consider the more general update $v_t = \frac{1}{t} \left(\sum_{j=1}^t g_j^2 + \varepsilon \mathbf{1} \right)$ similar to the previous case with AMSGRAD.

ADAMNC is analyzed in (Reddi et al., 2018, Theorem 5, Corollary 2) and similar to AMSGRAD it has been shown to exhibit $\mathcal{O}(\sqrt{T})$ worst case regret only when β_1 decreases to 0. We show in the following theorem that the same regret can be obtained with a constant β_1 .

Theorem 2. Under Assumption 1, $\beta_1 < 1$, and $\varepsilon > 0$, ADAMNC achieves the regret

$$R(T) \leq \frac{D^2\sqrt{T}}{2\alpha(1-\beta_1)} \sum_{i=1}^d v_{T,i}^{1/2} + \frac{2\alpha}{1-\beta_1} \sum_{i=1}^d \sqrt{\sum_{t=1}^T g_{t,i}^2}.$$

Algorithm 2 ADAMNC (Reddi et al., 2018)

- 1: **Input:** $x_1 \in \mathcal{X}$, $\alpha_t = \frac{\alpha}{\sqrt{t}}$, $\alpha > 0$, $\beta_1 < 1$, $\varepsilon \geq 0$, $m_0 = 0$.
 - 2: **for** $t = 1, 2 \dots$ **do**
 - 3: $g_t \in \partial f_t(x_t)$
 - 4: $m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t$
 - 5: $v_t = \frac{1}{t} \left(\sum_{j=1}^t g_j^2 + \varepsilon \mathbf{1} \right)$
 - 6: $x_{t+1} = P_{\mathcal{X}}^{v_t^{1/2}}(x_t - \alpha_t v_t^{-1/2} m_t)$
 - 7: **end for**
-

We skip the proof sketch of this theorem as it will have the same steps as AMSGRAD, just different estimation for $\alpha_t \|m_t\|_{v_t^{-1/2}}^2$, due to different v_t .

The full proof is given in the appendix.

Compared with the bound from (Reddi et al., 2018, Corollary 2), we see again that constant β_1 not only removes the middle term of (Reddi et al., 2018, Corollary 2) but improves the last term of the bound by a factor of $(1 - \beta_1)^2$.

3.5. SADAM

It is known that ADAGRAD can obtain logarithmic regret (Duchi et al., 2010), when the loss functions satisfy μ -strong convexity, defined as $f(x) \geq f(y) + \langle g, x - y \rangle + \frac{\mu}{2} \|y - x\|^2$, $\forall x, y \in \mathcal{X}$ and $g \in \partial f(y)$.

A variant of ADAMNC for this setting is proposed in (Wang et al., 2020, Theorem 1) and shown to obtain logarithmic regret, only with the assumption that β_1 decreases linearly to 0.

Algorithm 3 SADAM (Wang et al., 2020)

- 1: **Input:** $x_1 \in \mathcal{X}$, $\alpha_t = \frac{\alpha}{t}$, $\alpha > 0$, $\beta_1 < 1$, $m_0 = 0$, $\varepsilon \geq 0$, $\beta_{2t} = 1 - 1/t$.
 - 2: **for** $t = 1, 2 \dots$ **do**
 - 3: $g_t \in \partial f_t(x_t)$
 - 4: $m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t$
 - 5: $v_t = \beta_{2t} v_{t-1} + (1 - \beta_{2t})g_t^2$
 - 6: $\hat{v}_t = v_t + \frac{\varepsilon \mathbf{1}}{t}$
 - 7: $x_{t+1} = P_{\mathcal{X}}^{\hat{v}_t}(x_t - \alpha_t \hat{v}_t^{-1} m_t)$
 - 8: **end for**
-

Similar to AMSGRAD and ADAMNC, our new technique applies to SADAM to show logarithmic regret with a constant β_1 under the same assumptions as (Wang et al., 2020).

Theorem 3. *Let Assumption 1 hold and f_t be μ -strongly convex, $\forall t$. Then, if $\beta_1 < 1$, $\varepsilon > 0$, and $\alpha \geq \frac{G^2}{\mu}$, SADAM achieves*

$$R(T) \leq \frac{\beta_1 dGD}{1 - \beta_1} + \frac{\alpha}{1 - \beta_1} \sum_{i=1}^d \log \left(\frac{\sum_{t=1}^T g_{t,i}^2}{\varepsilon} + 1 \right).$$

Consistent with the standard literature of OGD (Hazan et al., 2007), to obtain the logarithmic regret, first step size α has a lower bound that depends on strong convexity constant μ . Compared with the requirement of (Wang et al., 2020) for $\alpha \geq \frac{G^2}{\mu(1-\beta_1)}$, our requirement is strictly milder as $1 - \beta_1 \leq 1$ and in practice since β_1 is near 1, it is much milder. We also remark that our bound is again strictly better than (Wang et al., 2020). Consistent with our previous results, we remove a factor of $\frac{1}{(1-\beta_1)^2}$ from the last term of the bound, compared to (Wang et al., 2020, Theorem 1).

We include the proof sketch to highlight how strong convexity helps in the analysis.

Proof sketch. We will start the same as proof sketch of Theorem 1 to get

$$\sum_{t=1}^T \langle g_t, x_t - x \rangle \leq S_1 + \frac{\beta_1}{1 - \beta_1} S_2 + \frac{\beta_1}{1 - \beta_1} S_3,$$

with the definitions of S_1, S_2, S_3 from the proof sketch of Theorem 1.

Now, due to strong convexity, one gets an improved estimate for the left-hand side,

$$\langle g_t, x_t - x \rangle \geq f_t(x_t) - f_t(x) + \frac{\mu}{2} \|x_t - x\|^2,$$

resulting in

$$R(T) \leq S_1 + \frac{\beta_1}{1 - \beta_1} S_2 + \frac{\beta_1}{1 - \beta_1} S_3 - \sum_{t=1}^T \frac{\mu}{2} \|x_t - x\|^2. \quad (7)$$

Similar as before, we note the bound for S_2 as

$$S_2 \leq \sum_{t=1}^{T-1} \alpha_t \|m_t\|_{\hat{v}_t^{-1/2}}^2. \quad (8)$$

For S_1 , one does not finish the estimation as before, but keep some terms that will be gotten rid of using strong convexity, and use the same estimation as S_2 to obtain

$$S_1 \leq \sum_{t=1}^T \sum_{i=1}^d \left(\frac{\hat{v}_{t,i}}{2\alpha_t} - \frac{\hat{v}_{t-1,i}}{2\alpha_{t-1}} \right) (x_{t,i} - x_i)^2 + \sum_{t=1}^T \frac{\alpha_t}{2} \|m_t\|_{\hat{v}_t^{-1/2}}^2. \quad (9)$$

As strong convexity gives more flexibility in the analysis, one can select $\alpha_t = \frac{\alpha}{t}$, resulting in an improved bound

$$\sum_{t=1}^T \frac{\alpha_t}{2} \|m_t\|_{\hat{v}_t^{-1/2}}^2 \leq \alpha \sum_{i=1}^d \log \left(\frac{\sum_{t=1}^T g_{t,i}^2}{\varepsilon} + 1 \right). \quad (10)$$

It is now easy to see that the negative term in (7), when first step size α is selected properly, can be used to remove the first term in the bound of (9).

It only remains to use Hölder inequality on S_3 , combine the estimates and use (10) to get the final bound. \square

4. Extensions

In this section, we further demonstrate the applicability of our analytic framework in different settings. First, we focus on the recently proposed zeroth-order version of AMSGRAD which required decreasing β_1 in the convex case (Chen et al., 2019b, Proposition 4), and we show that the same guarantees can be obtained with constant β_1 . Second, we show how to recover the known guarantees in the nonconvex setting, with small improvements. Finally, we extend our analysis to show that it allows any non-increasing variable β_{1t} schedule.

4.1. Zeroth order ADAM

We first recall the setting of (Chen et al., 2019b), where a zeroth order variant of AMSGRAD is proposed. The problem is

$$x_* \in \arg \min_{x \in \mathcal{X}} f(x) := \mathbb{E}_\xi [f(x; \xi)]. \quad (11)$$

We note that this stochastic optimization setting corresponds to a special case of general OCO, with independent and identically distributed loss functions $f(x; \xi)$, indexed by ξ .

The algorithm ZO-AdaMM (Chen et al., 2019b) is similar to AMSGRAD applied with a zeroth order gradient estimator \hat{g}_t , instead of regular gradient g_t . The gradient estimator is computed by

$$\hat{g}_t = (d/\mu) [f(x_t + \mu u; \xi_t) - f(x_t; \xi_t)] u, \quad (12)$$

where ξ_t is the sample selected at iteration t , u is a random vector drawn with uniform distribution from the sphere of a unit ball and μ is a sampling radius, or smoothing, parameter.

The benefit of this gradient estimator is that it is an unbiased estimator of the randomized smoothed version of f , i.e.,

$$f_\mu(x) = \mathbb{E}_{u \sim U_B} [f(x + \mu u)]. \quad (13)$$

From standard results in the zeroth-order optimization literature, it follows that $\mathbb{E}_u [\hat{g}_t] = \nabla f_\mu(x_t, \xi_t) = \nabla f_{t,\mu}(x_t)$. Moreover, for L_c -Lipschitz f and any $x \in \mathcal{X}$, we also have $\|f_\mu(x) - f(x)\| \leq \mu L_c$.

Two cases are analyzed by Chen et al. (2019b): convex f and nonconvex f . The authors proved guarantees with constant β_1 for nonconvex f (Chen et al., 2019b, Proposition 2).

However, surprisingly, their result for convex f requires $\beta_{1t} = \frac{\beta_1}{t}$ (Chen et al., 2019b, Proposition 4).

We identify that this discrepancy is due to the fact that their proof follows the same path as the standard regret analysis of Reddi et al. (2018). We give below a simple corollary of our technique showing that the same guarantees for convex f can be obtained with constant β_1 .

Proposition 1. *Assume that f is convex, L -smooth, and L_c -Lipschitz, \mathcal{X} is compact with diameter D . Then ZO-AdaMM with $\beta_1, \beta_2 < 1$, $\gamma = \frac{\beta_1^2}{\beta_2} < 1$ achieves*

$$\begin{aligned} \mathbb{E} \left[\sum_{t=1}^T f_{t,\mu}(x_t) - f_{t,\mu}(x_*) \right] \leq & \frac{D^2 \sqrt{T}}{2\alpha(1-\beta_1)} \sum_{i=1}^d \mathbb{E} \left[\hat{v}_{T,i}^{1/2} \right] \\ & + \frac{\alpha \sqrt{1 + \log T}}{\sqrt{(1-\beta_2)(1-\gamma)}} \sum_{i=1}^d \sqrt{\sum_{t=1}^T \mathbb{E} [\hat{g}_{t,i}^2]}. \end{aligned}$$

To finish the arguments, one can use standard bounds in zeroth order optimization, as in Chen et al. (2019b). Compared with Chen et al. (2019b, Proposition 4), the same remarks hold as for AMSGRAD. Not only our result allows constant β_1 , but it also comes with better constants.

4.2. Nonconvex AMSGRAD

In this section, we focus on the nonconvex, unconstrained, smooth, stochastic optimization setting:

$$\min_{x \in \mathbb{R}^d} f(x) := \mathbb{E}_\xi [f(x; \xi)].$$

More concretely, in this subsection we are working under the following assumption.

Assumption 2.

- $\triangleright f: \mathbb{R}^d \rightarrow \mathbb{R}$ is L -smooth, $G = \max_t \|\nabla f(x_t)\|_\infty$
- $\triangleright f_t(x) = f(x, \xi_t)$
- $\triangleright x_* \in \arg \min_x f(x)$ exists.

This is the only setting where theoretical guarantees with constant β_1 are known in the literature. We show in this section that our new analysis framework is not restricted to convex case, but it is flexible enough to also cover this case. We provide an alternative proof to those given in (Chen et al., 2019a; Zhou et al., 2018). Specifically, both proofs in (Chen et al., 2019a; Zhou et al., 2018) exploits the fact that, as $\mathcal{X} = \mathbb{R}^d$, there is no projection step in AMSGRAD. To handle first-order moment, these papers define an auxiliary iterate $z_t = x_t + \frac{\beta_1}{1-\beta_1}(x_t - x_{t-1})$, and invoke smoothness with z_{t+1} and z_t .

We give a different and simpler proof using our new analysis, without defining z_t . In terms of guarantees, we recover the same rates, with slightly better constants.

Theorem 4. *Under Assumption 2, $\beta_1 < 1$, $\beta_2 < 1$, and $\gamma = \frac{\beta_1^2}{\beta_2} < 1$ AMSGRAD achieves*

$$\begin{aligned} \frac{1}{T} \sum_{t=1}^T \mathbb{E} [\|\nabla f(x_t)\|^2] &\leq \frac{1}{\sqrt{T}} \left[\frac{G}{\alpha} (f(x_1) - f(x_*)) \right. \\ &\quad + \frac{G^3}{(1-\beta_1)} \|\hat{v}_0^{-1/2}\|_1 + \frac{G^3 d}{4L\alpha(1-\beta_1)} \\ &\quad \left. + \frac{2GLd\alpha(1-\beta_1)(1+\log T)}{(1-\beta_2)(1-\gamma)} \right]. \end{aligned}$$

Compared with (Chen et al., 2019a, Corollary 3.1), the initial value of $v_0 = \varepsilon$ only affects one of the terms in our bound, whereas $\frac{1}{\varepsilon}$ appears in all the terms of (Chen et al., 2019a, Corollary 3.1). The reason is that (Chen et al., 2019a) uses $v_0 \geq \varepsilon$ in many places of the proof, even when it was unnecessary.

Compared with (Zhou et al., 2018, Corollary 3.9), our result allows for bigger values of β_1 , since we require $\beta_1^2 \leq \beta_1 < 1$ whereas (Zhou et al., 2018, Corollary 3.9) requires $\beta_1 \leq \beta_2 < 1$. Moreover, (Zhou et al., 2018, Corollary 3.9) has a constant step size $\alpha = \frac{1}{\sqrt{dT}}$ that requires setting a horizon and becomes very small with large d .

Lastly, we have a $\log T$ dependence, whereas (Zhou et al., 2018, Corollary 3.9) does not. However, this is not for free and it stems from the choice of a constant step size $\alpha_t = \frac{1}{\sqrt{dt}}$ therein. In fact, it is well known that for online gradient descent analysis, $\log T$ can be shaved when $\alpha_t \approx \frac{1}{\sqrt{T}}$. However, in practice using a variable step size is more favorable, since it does not require setting T in advance. Therefore, we choose to work with variable step size and have the $\log T$ term in the bound.

4.3. Flexible β_1 schedules

We have focused on the case of constant β_1 throughout our paper, as it is the most popular choice in practice. However, it is possible that in some applications, practitioners might see benefit of using other schedules. For instance, one can decrease β_1 until some threshold and keep it constant afterwards. This is not covered by the previous regret analyses as β_1 needed to decrease to 0. With our framework however, one can use not only constant β_1 , but any schedule as long as it is nonincreasing, and optimal regret bounds will follow.

Due to space constraints, we do not repeat all the proofs with this modification, but illustrate the main change that happens with variable β_1 and show that our proofs will go through. In this section we switch to notation of β_{1t} to illustrate time-varying case.

We start from the result of Lemma 1, after summing over $t = 1, \dots, T$

$$\begin{aligned} \sum_{t=1}^T \langle g_t, x_t - x \rangle &= \sum_{t=1}^T \langle m_{t-1}, x_{t-1} - x \rangle \\ &\quad + \sum_{t=1}^T \frac{1}{1-\beta_{1t}} (\langle m_t, x_t - x \rangle - \langle m_{t-1}, x_{t-1} - x \rangle) \\ &\quad - \sum_{t=1}^T \frac{\beta_{1t}}{1-\beta_{1t}} \langle m_{t-1}, x_t - x_{t-1} \rangle. \end{aligned} \quad (14)$$

For bounding the terms on the first and third lines of (14), the only place that will change with varying β_{1t} in the proof, is that $\alpha_t \|m_t\|_{\hat{v}_t^{-1/2}}^2$ will have a slightly different estimation,

since now $m_t = \sum_{j=1}^t \prod_{k=1}^{t-j} \beta_{1(t-k+1)} (1-\beta_{1j}) g_j^2$. One can use that $\beta_{1t} \leq \beta_1$ to obtain the same bounds, but with $\frac{1}{(1-\beta_1)^2}$ factor multiplying the bounds now. As explained before, this is one thing we lose with varying β_{1t} in theory.

Next, we estimate the terms in the second line of (14)

$$\begin{aligned} \frac{1}{1-\beta_{1t}} (\langle m_t, x_t - x \rangle - \langle m_{t-1}, x_{t-1} - x \rangle) &= \\ \frac{1}{1-\beta_{1t}} \langle m_t, x_t - x \rangle - \frac{1}{1-\beta_{1(t-1)}} \langle m_{t-1}, x_{t-1} - x \rangle \\ &\quad + \left(\frac{\beta_{1(t-1)} - \beta_{1t}}{(1-\beta_{1t})(1-\beta_{1(t-1)})} \right) \langle m_{t-1}, x_{t-1} - x \rangle. \end{aligned}$$

Now, for the last line we use that β_{1t} is non-increasing, $\beta_{1t} \leq \beta_1$, $\|m_t\|_1 \leq dG$ and $\|x_t - x\|_\infty \leq D$, to get

$$\begin{aligned} \left(\frac{\beta_{1(t-1)} - \beta_{1t}}{(1-\beta_{1t})(1-\beta_{1(t-1)})} \right) \langle m_{t-1}, x_{t-1} - x \rangle \\ \leq \frac{dDG}{(1-\beta_1)^2} (\beta_{1(t-1)} - \beta_{1t}). \end{aligned} \quad (15)$$

Thus upon summation over $t = 1$ to T , as $m_0 = 0$,

$$\begin{aligned} \sum_{t=1}^T \frac{1}{1-\beta_{1t}} (\langle m_t, x_t - x \rangle - \langle m_{t-1}, x_{t-1} - x \rangle) &\leq \\ \frac{1}{1-\beta_T} \langle m_T, x_T - x \rangle + \frac{dDG}{(1-\beta_1)^2} (\beta_{10} - \beta_{1T}), \end{aligned} \quad (16)$$

where we let $\beta_{10} = \beta_{11} < 1$. Indeed, the contribution of this term will only be constant as $(1-\beta_{1t}) \leq 1, \forall t$, $\|m_t\|_\infty \leq G$, $\|x_t - x\|_\infty \leq D$.

Note that the estimation of the terms on the first and third lines of (14) are the same, as in the constant β_1 case (up to constants). Also, the contribution of the terms in the second line of (14) with varying β_{1t} is a constant. Thus, one can repeat our proofs, with any nonincreasing β_{1t} schedule and obtain the same optimal regret bounds, but with worse constants (compared to constant β_1 case).

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement no 725594 - time-data), the Swiss National Science Foundation (SNSF) under grant number 200021_178865/1, the Department of the Navy, Office of Naval Research (ONR) under a grant number N62909-17-1-211, the Hasler Foundation Program: Cyber Human Systems (project number 16066). PM acknowledges financial support from the French National Research Agency (ANR) under grant ORACLESS (ANR-16-CE33-0004-01) and the COST Action CA16229 “European Network for Game Theory” (GAMENET).

References

- Auer, P., Cesa-Bianchi, N., and Gentile, C. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002.
- Beckenbach, E. F. and Bellman, R. *Inequalities*, volume 30. Springer Science & Business Media, 1961.
- Chen, J. and Gu, Q. Closing the generalization gap of adaptive gradient methods in training deep neural networks. *arXiv preprint arXiv:1806.06763*, 2018.
- Chen, X., Liu, S., Sun, R., and Hong, M. On the convergence of a class of adam-type algorithms for non-convex optimization. In *International Conference on Learning Representations*, 2019a.
- Chen, X., Liu, S., Xu, K., Li, X., Lin, X., Hong, M., and Cox, D. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. In *Advances in Neural Information Processing Systems*, pp. 7202–7213, 2019b.
- Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. Technical Report UCB/Eecs-2010-24, Eecs Department, University of California, Berkeley, Mar 2010.
- Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.
- Fang, B. and Klabjan, D. Convergence analyses of online adam algorithm in convex setting and two-layer relu neural network. *arXiv preprint arXiv:1905.09356*, 2019.
- Hazan, E., Agarwal, A., and Kale, S. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.
- Hazan, E. et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.
- Huang, H., Wang, C., and Dong, B. Nostalgic adam: Weighting more of the past gradients when designing the adaptive learning rate. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 2556–2562, 2019.
- Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Li, X. and Orabona, F. On the convergence of stochastic gradient descent with adaptive stepsizes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 983–992, 2019.
- Luo, L., Xiong, Y., and Liu, Y. Adaptive gradient methods with dynamic bound of learning rate. In *International Conference on Learning Representations*, 2019.
- Reddi, S. J., Kale, S., and Kumar, S. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.
- Savarese, P. On the convergence of adabound and its connection to sgd. *arXiv preprint arXiv:1908.04457*, 2019.
- Wang, G., Lu, S., Cheng, Q., Tu, W., and Zhang, L. {SA}dam: A variant of adam for strongly convex functions. In *International Conference on Learning Representations*, 2020.
- Zhou, D., Tang, Y., Yang, Z., Cao, Y., and Gu, Q. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.
- Zou, F., Shen, L., Jie, Z., Zhang, W., and Liu, W. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11127–11135, 2019.