New efficient accelerated and stochastic gradient descent algorithms based on locally Lipschitz gradient constants

Nguyen Phung Hai Chung[1], **Pham Thi Hoai**[2,*], **Hoang Van Chung**[3]

[1,2,3]Hanoi University of Science and Technology

Email: hchung1997@gmail.com, hoai.phamthi@hust.edu.vn, chunghoangvan44@gmail.com

Abstract

In this paper, we revisit the recent stepsize used in the gradient descent scheme which is called NGD proposed by [Hoai et al., *A novel stepsize for gradient descent method*, Operations Research Letters (2024) 53, doi: 10.1016/j.orl.2024.107072]. We first investigate NGD stepsize with two well-known accelerated techniques which are Heavy ball and Nesterov's methods. In the convex setting of unconstrained nonlinear optimization problems, we show the ergodic convergence of the iterates obtained by accelerated versions of NGD with a sublinear rate. The stochastic versions of the proposed accelerated algorithms are introduced with analysis on the convergence in the nonconvex setting of the objective. Although our proposed algorithms require global Lipschitz continuity of the gradient, we do not utilize the global Lipschitz constant during computations. Instead, we leverage information about local Lipschitz constants derived from previous iterations. Numerical experiments on numerous practical problems in machine learning and deep learning problems demonstrate the efficiency of our proposed methods compared to the recent ones.

Key words. Stochastic gradient descent, Nesterov's acceleration, Heavy Ball, adaptive stepsize, machine learning, accelerated method.

AMS subject classifications. 90C25, 90C15, 65K05, 68T05.

1 Introduction

Unconstrained nonlinear optimization problems have received a lot of attention from researchers since they have many applications in economics, data science, machine learning, deep learning, etc, see e.g. [1, 2] and the references therein. The formulation of this problem is the following

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is smooth. Leveraging the differentiability of the objective function f, first-order methods have been studied widely for solving Problem (1) in the literature, see [3, 4, 5, 6, 7]. Among these methods, gradient descent plays an important role due to its ease of implementation and efficient performance. Gradient descent was originally proposed by Cauchy [8] in 1847 and has become classical. Starting at some point $x_0 \in \mathbb{R}^n$, this method uses the idea of updating the variable $x_k \in \mathbb{R}^n$ at each iteration $k \in \mathbb{N}$ by the formula

$$x_{k+1} = x_k - \lambda_k \nabla f(x_k), \tag{GD}$$

where $\lambda_k > 0$ is the stepsize at iteration k. The usual condition of f for guaranteeing the convergence of gradient scheme (GD) is the global Lipschitzness of the gradient ∇f over $\mathbb{R}^n$, i.e., there exists a constant $L > 0$ such that

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

When f is convex and λ_k is defined as a fixed number in $\left(0, \frac{1}{L}\right]$ or λ_k is computed by using backtracking line search procedures, the convergence rate of (GD) is $O\left(\frac{1}{k}\right)$ for $f(x^k) - f_*$, see e.g. [6, 7]. However, estimating the global constant L is often impractical, and backtracking linesearch can be computationally expensive. Furthermore, both strategies can suffer from small stepsizes caused by a large L or frequent backtracking calculations. To address these drawbacks, adaptive stepsize selections used for (GD) have been proposed by Malitsky and Mischenko [9] and Hoai et al. [10] recently. Particularly, AdGD given by [9] determines

$$\lambda_k = \min\left\{\sqrt{1+\theta_{k-1}}\lambda_{k-1}, \frac{\|x_k - x_{k-1}\|}{2\|\nabla f(x_k) - \nabla f(x_{k-1})\|}\right\}, \quad k \geq 1, \tag{2}$$

where $\theta_0 = +\infty, \theta_k = \lambda_k/\lambda_{k-1}$; and NGD provided by [10] updates λ_k as follows: for $0 < \eta_1 < \eta_0 < 1/2$, and a convergence positive series $\sum_{k=0}^{+\infty} \varepsilon_k$,

$$\lambda_k = \begin{cases} \eta_1 \frac{\|x_k - x_{k-1}\|}{\|\nabla f(x_k) - \nabla f(x_{k-1})\|}, & \text{if } \|\nabla f(x_k) - \nabla f(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\| \\ (1+\varepsilon'_{k-1})\lambda_{k-1}, & \text{otherwise} \end{cases} \tag{3}$$

where $\varepsilon'_k = \begin{cases} \varepsilon_k & \text{if } \frac{\lambda_{k-1}}{\lambda_{k-2}} \geq 1 \\ \varepsilon'_{k-1} = \min\left\{\varepsilon_{k-1}, \sqrt{1+\frac{\lambda_{k-1}}{\lambda_{k-2}}} - 1\right\} & \text{otherwise} \end{cases}$. Both of AdGD and NGD re-

quire neither backtracking line search procedure nor the information of Lipschitz constant L and they utilize the information of locally Lipschitz constant in deriving the stepsize. It is also worth noting that AdGD and NGD work without using the global Lipschitz condition on ∇f, they only require ∇f being locally Lipschitz continuous.

In order to improve the speed of (GD) scheme, one can use accelerated techniques such as the two classical accelerated techniques introduced by Polyak [11, 3] and Nesterov [4]. In particular, Heavy ball technique given in [11, 3] takes

$$x_{k+1} = x_k - \lambda_k \nabla f(x_k) + \gamma(x_k - x_{k-1}). \qquad \text{(Heavy ball method)}$$

and Nesterov's acceleration in [4] determines

$$\begin{aligned} y_{k+1} &= x_k - \lambda_k \nabla f(x_k) \\ x_{k+1} &= y_{k+1} + \gamma(y_{k+1} - y_k). \end{aligned} \qquad \text{(Nesterov's method)}$$

Heavy ball method and Nesterov's method are two-step methods because x_{k+1} is computed via x_k and x_{k-1}. The constants $\gamma > 0$ in Heavy ball method and Nesterov's method are called accelerated factors.

Contributions: Our contributions in this paper are the followings:

(i) Firstly, we solve an open question given by Hoai et al. [10] by proposing accelerated versions of NGD. Particularly, two new algorithms **NGDh** and **NGDn** combining NGD with two accelerated techniques Heavy ball method and Nesterov's method, respectively, are proposed in this paper.

From a **theoretical perspective**, our convergence results are comparable to those of other methods in the literature that employ Heavy ball method and Nesterov's method. In particular,

- In the case of convex f, we prove the *ergodic convergence* of **NGDn** and **NGDh** with the rate $O(\frac{1}{K})$ from some fixed iteration. This convergence result is similar to that of the Heavy-ball method given in [12] which uses the fixed stepsize selection λ_k (which requires knowing aL). Compared with the convergence rate $O(\frac{1}{k^2})$ of Nesterov's method in [4] (with $\lambda_k = \frac{1}{L}$ or λ_k computed by backtracking line search), ours is theoretically weaker. But **in practice**, **NGDh** and **NGDn** use *adaptive stepsizes* λ_k that is easy to implement without estimating the Lipschitz constant L or suffering from the expensive computation of backtracking procedures.
- It is worth noting that in the literature AdGD-accel given in [9] use Nesterov's acceleration with adaptive stepsize AdGD but its convergence has not yet been provided.

(ii) Our **second contribution** is dedicated to studying the stochastic versions of **NGDh** and

NGDn. We proposed two new stochastic algorithms named **SNGDh** and **SNGDn**. It is well known that, the stochastic approach is useful for Problem (1) in machine learning and deep learning (see e.g., [13] and the references therein) when working with big data. This is because it overcomes the expensive costs of the deterministic way that requires the full computation of the gradient at each iteration.

From a theoretical point of view, we obtain typical convergence results for our stochastic algorithms SNGDn and SNGDh under standard conditions. In particular,

- Under the classical conditions including: 1. the uniform boundedness of the stochastic gradients (see e.g., [14, 15, 16]) and 2. the globally Lipchitz gradient condition of the *nonconvex* objective function f; we prove that the best expected squared norm of the gradient is bounded. This ensures that *the best iterates obtained by our proposed stochastic algorithms are close to some stationary points of the objective function.*
- In comparison with the related AdSGD algorithm [9] (the stochastic version of AdGD), our convergence results for SNGDn and SNGDh are obtained for nonconvex functions f, which are popular in machine learning, whereas AdSGD's convergence is obtained for strongly convex f.

(iii) From a practical standpoint, the advantages of the adaptive step size based on the local Lipschitz constant - used in our new algorithms NGDn, NGDh, SNGDh, and SNGDn - provide significantly more efficient performance compared to other recent state-of-the-art methods. This is demonstrated by numerical results for numerous benchmark problems in machine learning and deep learning in Section 5.

The rest of the paper is organized as follows. After recalling some fundamental results in Section 2 we propose the new accelerated algorithms in Section 3 with the convergent analysis. The stochastic versions of algorithms in Section 3 together with their convergences are presented in Section 4. Finally, the numerical results are reported in Section 5 with available codes in our repository https://github.com/hoaiphamthi/Accelerated-and-Stochastic-NGD.

2 Preliminaries

Throughout this paper, our underlying space is $\mathbb{R}^n$ equipped with the standard Euclidean norm, denoted by $\|x\| = \sqrt{\langle x, x\rangle}$.

Definition 2.1. A function $f\colon \mathbb{R}^n \to \mathbb{R}$ is called **convex** if

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y) \quad \text{for all } x, y \in \mathbb{R}^n,\ \lambda \in [0,1].$$

Definition 2.2. A function $f\colon \mathbb{R}^n \to \mathbb{R}$ is called L-smooth if it is differentiable and its gradient ∇f is L-Lipschitz continuous, i.e.,

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\| \quad \text{for all } x, y \in \mathbb{R}^n.$$

The following lemma recalls some fundamental properties of L-smooth functions.

Lemma 2.1. *If f is L-smooth on $\mathbb{R}^n$, then*

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2 \quad \textit{for all } x, y \in \mathbb{R}^n.$$

Furthermore, if f is also convex, the following inequality holds:

$$f(x) - f(y) + \frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2 \leq \langle x - y, \nabla f(x) \rangle \quad \textit{for all } x, y \in \mathbb{R}^n.$$

Proof. We refer the reader to [5, 4] for the detailed proofs. □

Lemma 2.2. *Let $a \in [0, 1)$ and let i, Q be integers such that $Q \geq i$. Then,*

$$\sum_{q=i}^{Q} a^q q \leq \frac{a}{(1-a)^2}.$$

Proof. We refer the readers to [17] for the detailed proof. □

We consider the optimization problem (1) under the following assumption which hold throughout the paper in both deterministic and stochastic settings:

Assumption 2.1. *The set of optimal solutions of Problem* (1)*, denoted by X^*, is nonempty. We denote by f_* the optimal value of the objective function, i.e., $f_* = f(x^*)$ for any $x^* \in X^*$.*

For specific settings in the convergence analysis, any further assumptions will be clarified explicitly at the beginning of the respective sections.

3 New accelerated gradient descent algorithms

In this section, we propose two accelerated algorithms that incorporate the Heavy Ball [11] and Nesterov [4] momentum techniques into the NGD adaptive stepsize framework [10]. Unlike standard methods that rely on a global Lipschitz constant, our approach adapts the stepsize based on local curvature information approximated by finite differences of gradients.

Algorithm 1 NGD accelerated by Heavy ball method (NGDh)

1: **Initialization.** Select $\lambda_0 > 0$, $0 < \eta_1 < \eta_0$, $0 \leq \gamma < 1$ and a positive real sequence $\{\varepsilon_k\}$ such that $\sum_{k=0}^{+\infty} \varepsilon_k < +\infty$. Choose $x_0 \in \mathbb{R}^n$, $x_1 = x_0 - \lambda_0 \nabla f(x_0)$.
2: **for** $k = 1, 2, \ldots$ **do**
3: **if** $\|\nabla f(x_k) - \nabla f(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}} \|x_k - x_{k-1}\|$ **then**
4: $\lambda_k = \eta_1 \frac{\|x_k - x_{k-1}\|}{\|\nabla f(x_k) - \nabla f(x_{k-1})\|}$
5: **else**
6: $\lambda_k = (1 + \varepsilon_{k-1}) \lambda_{k-1}$
7: **end if**
8: $x_{k+1} = x_k - \lambda_k \nabla f(x_k) + \gamma(x_k - x_{k-1})$
9: **end for**

Algorithm 2 NGD accelerated by Nesterov's method (NGDn)

1: **Initialization.** Select $\lambda_0 > 0$, $0 < \eta_1 < \eta_0$, $0 \leq \gamma < 1$ and a positive real sequence $\{\varepsilon_k\}$ such that $\sum_{k=0}^{+\infty} \varepsilon_k < +\infty$. Choose $x_0 \in \mathbb{R}^n$, $x_1 = x_0 - \lambda_0 \nabla f(x_0)$.
2: **for** $k = 1, 2, \ldots$ **do**
3: **if** $\|\nabla f(x_k) - \nabla f(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}} \|x_k - x_{k-1}\|$ **then**
4: $\lambda_k = \eta_1 \frac{\|x_k - x_{k-1}\|}{\|\nabla f(x_k) - \nabla f(x_{k-1})\|}$
5: **else**
6: $\lambda_k = (1 + \varepsilon_{k-1}) \lambda_{k-1}$
7: **end if**
8: $y_{k+1} = x_k - \lambda_k \nabla f(x_k)$
9: $x_{k+1} = y_{k+1} + \gamma(y_{k+1} - y_k)$
10: **end for**

It is observed that our proposed algorithms, Algorithm 1 (NGDh) and Algorithm 2 (NGDn), dynamically update the stepsize λ_k. If the local curvature condition (Line 4) is violated, the stepsize is reduced inversely proportional to the local Lipschitz estimate. Otherwise, it is slightly increased to accelerate convergence.

Remark 3.1. Algorithm 1 is the Heavy-ball version of NGD, while Algorithm 2 incorporates Nesterov's acceleration. It is easy to see that in the case where $\gamma = 0$ and $0 < \eta_1 < \eta_0 < \frac{1}{2}$, Algorithms 1 and 2 are similar to Algorithm 2.1 (NGD) in [10]. The range for η_0 and η_1 in Algorithms 1 and 2 is now enlarged to $(0, +\infty)$, compared to $(0, \frac{1}{2})$ for Algorithm 2.1 in [10]. Additionally, unlike Algorithm 2.1, we do not need to update ε_k in Algorithms 1 and 2. These changes make the accelerated versions of NGD simpler and could increase the step lengths.

Next, we will investigate the convergence of **NGDh** and **NGDn** with additional assumptions on f below.

Assumption 3.1. *The objective function f is L-smooth on $\mathbb{R}^n$.*

Assumption 3.2. *The objective function f is convex on $\mathbb{R}^n$.*

Before establishing the convergence of Algorithms 1 and 2, it is essential to examine the behavior of the adaptive stepsize. The following lemma, which is analogous to Lemma 2.3 in [10], asserts that the sequence $\{\lambda_k\}$ is bounded and converges to a positive limit.

Lemma 3.1. *Let $\{\lambda_k\}$ be the sequence generated by Algorithm 1 or Algorithm 2. If f satisfies Assumptions 2.1 and 3.1, then:*

(i) For all $k \geq 0$, we have

$$\lambda_k \geq \min\left\{\lambda_0, \frac{\eta_1}{L}\right\}. \tag{4}$$

(ii) The sequence $\{\lambda_k\}$ converges to a limit $\overline{\lambda} < +\infty$.

Proof. Note that both of Algorithm 1 and 2 include the same update rule for λ_k (lines 2-6 of each algorithm) hence in the arguments of this proof we apply for Algorithm 1, those of Algorithm 2 are similar.

(i) Clearly, inequality (4) holds for $k = 0$. For $k \geq 1$, we consider two cases.

Case 1: If $\|\nabla f(x_k) - \nabla f(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\|$, then

$$\lambda_k = \frac{\eta_1\|x_k - x_{k-1}\|}{\|\nabla f(x_k) - \nabla f(x_{k-1})\|} \geq \frac{\eta_1}{L},$$

which follows from the L-smoothness of f.

Case 2: If $\|\nabla f(x_k) - \nabla f(x_{k-1})\| \leq \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\|$, then

$$\lambda_k = (1 + \varepsilon_{k-1})\lambda_{k-1} \geq \lambda_{k-1}.$$

By induction, we conclude that $\lambda_k \geq \min\left\{\lambda_0, \frac{\eta_1}{L}\right\}$ for all $k \geq 0$.

(ii) From Algorithm 1, it follows that

$$a_k = \ln\left(\frac{\lambda_{k+1}}{\lambda_k}\right) \leq \ln(1 + \varepsilon_k) \leq \varepsilon_k, \quad \forall k \geq 0.$$

Decompose $a_k = a_k^+ - a_k^-$, where $a_k^+ = \max(0, a_k)$ and $a_k^- = -\min(0, a_k)$. Consequently, $a_k^+ \geq 0$ and $a_k^- \geq 0$ for all $k \geq 0$. Moreover, since $a_k^+ \leq \varepsilon_k$, the convergence of $\sum_{k=0}^{+\infty} \varepsilon_k$ implies the convergence of $\sum_{k=0}^{+\infty} a_k^+$.

Now, consider the partial sum:

$$\sum_{i=0}^{k} a_i = \ln(\lambda_{k+1}) - \ln(\lambda_0) = \sum_{i=0}^{k}(a_i^+ - a_i^-) = \sum_{i=0}^{k} a_i^+ - \sum_{i=0}^{k} a_i^-. \tag{5}$$

If $\lim_{k\to+\infty}\sum_{i=0}^{k} a_i^- = +\infty$, then $\lim_{k\to+\infty}\ln(\lambda_k) = -\infty$, which is equivalent to $\lim_{k\to+\infty}\lambda_k = 0$. This contradicts Lemma 3.1(i), which states that $\lambda_k \geq \min\left\{\lambda_0, \frac{\eta_1}{L}\right\} > 0$ for all $k \geq 0$. Therefore, the series $\sum_{k=0}^{+\infty} a_k^-$ is convergent. From (5), we conclude that $\lim_{k\to+\infty}\lambda_k = \overline{\lambda} < +\infty$.

□

Lemma 3.2. *Suppose that f satisfies Assumption 3.1. Let $\{\lambda_k\}$ be the sequence generated by Algorithm 1 (or Algorithm 2). Then, there exists an integer $\overline{k}$ such that*

$$\|\nabla f(x_k) - \nabla f(x_{k-1})\| \leq \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\|, \quad \forall k \geq \overline{k}. \tag{6}$$

Consequently, $\lambda_{k-1} \leq \lambda_k \leq \overline{\lambda} = \lim_{j\to+\infty}\lambda_j$ for all $k \geq \overline{k}$.

Proof. Suppose, for the sake of contradiction, that there exists a subsequence $\{k_j\}$ with $k_j \to +\infty$ such that

$$\|\nabla f(x_{k_j}) - \nabla f(x_{k_j-1})\| > \frac{\eta_0}{\lambda_{k_j-1}}\|x_{k_j} - x_{k_j-1}\|.$$

In this case, the algorithm updates λ_{k_j} as:

$$\lambda_{k_j} = \eta_1\frac{\|x_{k_j} - x_{k_j-1}\|}{\|\nabla f(x_{k_j}) - \nabla f(x_{k_j-1})\|}.$$

This implies

$$\frac{\eta_1\|x_{k_j} - x_{k_j-1}\|}{\lambda_{k_j}} = \|\nabla f(x_{k_j}) - \nabla f(x_{k_j-1})\| > \frac{\eta_0}{\lambda_{k_j-1}}\|x_{k_j} - x_{k_j-1}\|.$$

Therefore, we have $\frac{\lambda_{k_j}}{\lambda_{k_j-1}} < \frac{\eta_1}{\eta_0}$ for all k_j. On the other hand, from Lemma 3.1, we know that $\lim_{k_j\to+\infty}\lambda_{k_j} = \lim_{k_j\to+\infty}\lambda_{k_j-1} = \lim_{k\to+\infty}\lambda_k = \overline{\lambda}$. Taking the limit, we deduce that

$$\frac{\overline{\lambda}}{\overline{\lambda}} \leq \frac{\eta_1}{\eta_0} < 1,$$

which implies $1 < 1$. This is a contradiction, completing the proof. □

Lemma 3.2 just establishes the relationship between the local Lipschitz approximation of ∇f and the stepsize for sufficiently large iterations. Furthermore, it shows that the sequence of stepsizes becomes monotonically increasing and converges to a finite limit after a certain iteration.

3.1 The convergence of Algorithm 1 (NGDh)

In this subsection, we establish the convergence rate of the NGDh algorithm. We show that the ergodic average of the iterates converges to the optimal value at a sublinear rate.

Theorem 3.1. *Consider Problem (1) under Assumptions 2.1, 3.1, and 3.2. Let $\overline{k}$ be the iteration index defined in Lemma 3.2. Suppose the parameters satisfy the condition:*

$$e^{\sum_{j=\overline{k}}^{+\infty}\varepsilon_{j-1}}\frac{\eta_1\|x_{\overline{k}}-x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}})-\nabla f(x_{\overline{k}-1})\|}\leq\frac{1-\gamma}{L}. \tag{7}$$

Then, for any $K>\overline{k}$, the weighted average iterate $\overline{x}_K$ defined by

$$\overline{x}_K=\frac{\gamma x_K+\sum_{k=1}^{K}x_k}{\gamma+(1-\gamma)K}$$

satisfies the following convergence bound:

$$f(\overline{x}_K)-f_*\leq\frac{C}{\gamma+(1-\gamma)K}=O\left(\frac{1}{K}\right),$$

where C is a positive constant depending on the initialization and the function values at the first $\overline{k}$ steps.

Proof. From the update rule of Algorithm 1, for each iteration $k\geq 1$, we have:

$$x_{k+1}-\gamma x_k=x_k-\gamma x_{k-1}-\lambda_k\nabla f(x_k). \tag{8}$$

Therefore, for any $x^*\in X^*$, we evaluate the expression:

$$\begin{aligned}\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2&=\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2+\lambda_k^2\|\nabla f(x_k)\|^2\\&\quad-2\langle x_k-\gamma x_{k-1}-(1-\gamma)x^*,\lambda_k\nabla f(x_k)\rangle\\&=\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2+\lambda_k^2\|\nabla f(x_k)\|^2\\&\quad-2\langle(1-\gamma)(x_k-x^*)+\gamma(x_k-x_{k-1}),\lambda_k\nabla f(x_k)\rangle\\&=\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2+\lambda_k^2\|\nabla f(x_k)\|^2\\&\quad-2(1-\gamma)\lambda_k\langle x_k-x^*,\nabla f(x_k)\rangle-2\gamma\lambda_k\langle x_k-x_{k-1},\nabla f(x_k)\rangle.\end{aligned} \tag{9}$$

Since f is L-smooth and convex, we apply inequality (2.1) from Lemma 2.1 to (9):

$$\begin{aligned}\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2&\leq\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2+\lambda_k^2\|\nabla f(x_k)\|^2\\&\quad-2(1-\gamma)\lambda_k\left(f(x_k)-f_*+\frac{1}{2L}\|\nabla f(x_k)\|^2\right)\\&\quad-2\gamma\lambda_k\left(f(x_k)-f(x_{k-1})+\frac{1}{2L}\|\nabla f(x_k)-\nabla f(x_{k-1})\|^2\right)\\&\leq\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2-2(1-\gamma)\lambda_k\left(f(x_k)-f_*\right)\\&\quad-2\gamma\lambda_k\left(f(x_k)-f(x_{k-1})\right)+\left(\lambda_k^2-\frac{(1-\gamma)\lambda_k}{L}\right)\|\nabla f(x_k)\|^2.\end{aligned} \tag{10}$$

Dividing by $2\lambda_k$, we obtain:

$$\frac{1}{2\lambda_k}\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2 \le \frac{1}{2\lambda_k}\|x_k-\gamma x_{k-1}-(1-\gamma)x^*\|^2-(1-\gamma)\left(f(x_k)-f_*\right) \\ -\gamma\left(f(x_k)-f(x_{k-1})\right)+\frac{1}{2}\left(\lambda_k-\frac{1-\gamma}{L}\right)\|\nabla f(x_k)\|^2. \quad (11)$$

Summing (11) from $k=1$ to K (where $K>\overline{k}$), we get:

$$\sum_{k=1}^{K}\left((1-\gamma)(f(x_k)-f_*)+\gamma(f(x_k)-f(x_{k-1}))\right) \le \frac{1}{2\lambda_1}\|x_1-\gamma x_0-(1-\gamma)x^*\|^2 \\ +\sum_{k=1}^{K-1}\left(\frac{1}{2\lambda_{k+1}}-\frac{1}{2\lambda_k}\right)\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2 \\ -\frac{1}{2\lambda_K}\|x_{K+1}-\gamma x_K-(1-\gamma)x^*\|^2 \\ +\frac{1}{2}\sum_{k=1}^{K}\left(\lambda_k-\frac{1-\gamma}{L}\right)\|\nabla f(x_k)\|^2. \quad (12)$$

Since $\overline{k}$ is the smallest integer satisfying inequality (6) in Lemma 3.2, the following properties hold:

- First, $\lambda_k \le \lambda_{k+1}$ for all $k \ge \overline{k}$, which implies:

$$\sum_{k=\overline{k}}^{K-1}\left(\frac{1}{2\lambda_{k+1}}-\frac{1}{2\lambda_k}\right)\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2 \le 0. \quad (13)$$

- Second, since $\lambda_{\overline{k}-1}=\frac{\eta_1\|x_{\overline{k}}-x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}})-\nabla f(x_{\overline{k}-1})\|}$, it follows from (7) that for all $k \ge \overline{k}$:

$$\lambda_k=(1+\varepsilon_{k-1})\lambda_{k-1}=\ldots=\prod_{j=\overline{k}}^{k}(1+\varepsilon_{j-1})\lambda_{\overline{k}-1} \le e^{\sum_{j=\overline{k}}^{k}\varepsilon_{j-1}}\lambda_{\overline{k}-1} \\ \le e^{\sum_{j=\overline{k}}^{+\infty}\varepsilon_{k-1}}\frac{\eta_1\|x_{\overline{k}}-x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}})-\nabla f(x_{\overline{k}-1})\|} \le \frac{1-\gamma}{L}. \quad (14)$$

Substituting (13) and (14) into (12), we obtain:

$$(1-\gamma)\sum_{k=1}^{K}(f(x_k)-f_*)+\gamma(f(x_K)-f(x_0)) \le \sum_{k=1}^{\overline{k}-1}\left(\frac{1}{2\lambda_{k+1}}-\frac{1}{2\lambda_k}\right)\|x_{k+1}-\gamma x_k-(1-\gamma)x^*\|^2 \\ +\frac{1}{2\lambda_1}\|x_1-\gamma x_0-(1-\gamma)x^*\|^2+\frac{1}{2}\sum_{k=1}^{\overline{k}-1}\left(\lambda_k-\frac{1-\gamma}{L}\right)\|\nabla f(x_k)\|^2. \quad (15)$$

Let us define the constant C as:

$$C = \gamma(f(x_0) - f_*) + \sum_{k=1}^{\overline{k}-1} \left(\frac{1}{2\lambda_{k+1}} - \frac{1}{2\lambda_k} \right) \|x_{k+1} - \gamma x_k - (1-\gamma)x^*\|^2$$
$$+ \frac{1}{2\lambda_1} \|x_1 - \gamma x_0 - (1-\gamma)x^*\|^2 + \frac{1}{2} \sum_{k=1}^{\overline{k}-1} \left(\lambda_k - \frac{1-\gamma}{L} \right) \|\nabla f(x_k)\|^2.$$

Then, inequality (15) can be rewritten as:

$$\gamma(f(x_K) - f_*) + (1-\gamma) \sum_{k=1}^{K} (f(x_k) - f_*) \leq C. \tag{16}$$

By the convexity of f, we have:

$$f\left(\frac{\gamma x_K + (1-\gamma) \sum_{k=1}^{K} x_k}{\gamma + (1-\gamma)K} \right) \leq \frac{\gamma f(x_K) + \sum_{k=1}^{K} f(x_k)}{\gamma + (1-\gamma)K}. \tag{17}$$

Combining (16) and (17), we conclude that:

$$f(\overline{x}_K) - f_* \leq \frac{C}{\gamma + (1-\gamma)K} = O\left(\frac{1}{K} \right), \quad \forall K > \overline{k}, \tag{18}$$

where

$$\overline{x}_K = \frac{\gamma x_K + \sum_{k=1}^{K} x_k}{\gamma + (1-\gamma)K}.$$

□

Remark 3.2. Following the classical scheme in [3], λ_k is chosen as a fixed constant (i.e., $\lambda_k = \lambda \in (0, 2(1+\gamma)/L)$ for all $k = 0, 1, ...$) and the corresponding *local convergence* result of Heavy ball method was obtained for f satisfying: 1. μ-strong convexity, 2. twice differentiable, 3. having a global Lipschitz gradient. Recently, in [12], the authors provided the *global convergence* of Heavy ball method for convex f and fixed stepsize λ_k. They obtained the ergodic convergence with the sublinear rate $O(\frac{1}{k})$. Hence our convergence result obtained in Theorem 3.1 is comparable with those in literature and has advantage in eliminating the need to know the global Lipschitz constant L for computing λ_k.

3.2 The convergence of Algorithm 2 (NGDn)

We now establish the ergodic convergence of the Nesterov-accelerated version (NGDn). Similar to the previous section, the convergence relies on the eventual stability of the stepsize.

Theorem 3.2. *Consider Problem (1) under Assumptions 2.1, 3.1, and 3.2. Let* $\overline{k}$ *be the*

smallest integer satisfying the condition in Lemma 3.2. Suppose the parameters satisfy:

$$e^{\sum_{j=\overline{k}}^{+\infty}\varepsilon_{j-1}}\frac{\eta_1\|x_{\overline{k}}-x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}})-\nabla f(x_{\overline{k}-1})\|}\leq\min\left\{\frac{1-\gamma}{L}+\gamma\lambda_{min},\frac{1}{L}\right\}. \tag{19}$$

Then, for any $K>\overline{k}$, the sequence $\{x_k\}$ generated by Algorithm 2 satisfies:

$$f(\overline{x}_K)-f_*\leq\frac{D}{K(1-\gamma)+\gamma}=O\left(\frac{1}{K}\right),$$

where D is a positive constant and the ergodic iterate is defined as $\overline{x}_K=\dfrac{\gamma x_K+\sum_{k=1}^{K}x_k}{\gamma+(1-\gamma)K}$.

Proof. From the update rules of Algorithm 2, we have for $k\geq 0$ (with $y_1=y_0$):

$$\begin{aligned}y_{k+1}&=x_k-\lambda_k\nabla f(x_k),\\ x_{k+1}&=y_{k+1}+\gamma(y_{k+1}-y_k).\end{aligned} \tag{20}$$

Let us define $a_0=\frac{\gamma}{1-\gamma}\lambda_0\nabla f(x_0)$ and

$$a_{k+1}=\frac{\gamma}{1-\gamma}\left(x_{k+1}-x_k+\lambda_k\nabla f(x_k)\right),\quad\text{for } k\geq 0. \tag{21}$$

It is straightforward to verify that

$$x_{k+1}+a_{k+1}=x_{k+1}+\frac{\gamma}{1-\gamma}\left(x_{k+1}-x_k+\lambda_k\nabla f(x_k)\right)=\frac{x_{k+1}}{1-\gamma}+\frac{\gamma}{1-\gamma}(\lambda_k\nabla f(x_k)-x_k).$$

From (20), we can write $x_{k+1}=(1+\gamma)\left(x_k+\lambda_k\nabla f(x_k)\right)-\gamma\left(x_{k-1}-\lambda_{k-1}\nabla f(x_{k-1})\right)$. Hence, for all $k\geq 0$:

$$\begin{aligned}x_{k+1}+a_{k+1}&=\frac{1+\gamma}{1-\gamma}\left(x_k-\lambda_k\nabla f(x_k)\right)-\frac{\gamma}{1-\gamma}\left(x_{k-1}-\lambda_{k-1}\nabla f(x_{k-1})\right)+\frac{\gamma}{1-\gamma}\left(\lambda_k\nabla f(x_k)-x_k\right)\\ &=x_k+\frac{\gamma}{1-\gamma}x_k-\frac{\lambda_k\nabla f(x_k)}{1-\gamma}-\frac{\gamma}{1-\gamma}\left(x_{k-1}-\lambda_{k-1}\nabla f(x_{k-1})\right)\\ &=x_k+\frac{\gamma}{1-\gamma}\left(x_k-x_{k-1}+\lambda_{k-1}\nabla f(x_{k-1})\right)-\frac{\lambda_k\nabla f(x_k)}{1-\gamma}\\ &=x_k+a_k-\frac{\lambda_k\nabla f(x_k)}{1-\gamma}.\end{aligned}$$

Squaring the norm of both sides yields:

$$\begin{aligned}\|x_{k+1}+a_{k+1}-x^*\|^2 &= \left\|x_k+a_k-\frac{\lambda_k\nabla f(x_k)}{1-\gamma}-x^*\right\|^2\\ &= \|x_k+a_k-x^*\|^2+\frac{\lambda_k^2}{(1-\gamma)^2}\|\nabla f(x_k)\|^2-\frac{2\lambda_k}{1-\gamma}\langle x_k+a_k-x^*,\nabla f(x_k)\rangle\\ &= \|x_k+a_k-x^*\|^2+\frac{\lambda_k^2}{(1-\gamma)^2}\|\nabla f(x_k)\|^2-\frac{2\lambda_k}{1-\gamma}\langle x_k-x^*,\nabla f(x_k)\rangle\\ &\quad-\frac{2\gamma\lambda_k}{(1-\gamma)^2}\langle x_k-x_{k-1},\nabla f(x_k)\rangle-\frac{2\gamma\lambda_k\lambda_{k-1}}{(1-\gamma)^2}\langle\nabla f(x_{k-1}),\nabla f(x_k)\rangle.\end{aligned}$$

By the L-smoothness and convexity of f, and applying Lemma 2.1 (specifically inequality (2.1)), we obtain the following bound:

$$\begin{aligned}\|x_{k+1}+a_{k+1}-x^*\|^2 &\le \|x_k+a_k-x^*\|^2-\frac{2\lambda_k}{1-\gamma}(f(x_k)-f_*)-\frac{\lambda_k}{(1-\gamma)L}\|\nabla f(x_k)\|^2\\ &\quad+\frac{\lambda_k^2\|\nabla f(x_k)\|^2}{(1-\gamma)^2}-\frac{2\gamma\lambda_k}{(1-\gamma)^2}(f(x_k)-f(x_{k-1}))\\ &\quad-\frac{\gamma\lambda_k}{L(1-\gamma)^2}\|\nabla f(x_k)-\nabla f(x_{k-1})\|^2-\frac{2\gamma\lambda_k\lambda_{k-1}}{(1-\gamma)^2}\langle\nabla f(x_{k-1}),\nabla f(x_k)\rangle.\end{aligned}\tag{22}$$

Using the identity $-2\langle a,b\rangle=\|a-b\|^2-\|a\|^2-\|b\|^2$, inequality (22) can be rewritten as:

$$\begin{aligned}\|x_{k+1}+a_{k+1}-x^*\|^2 &\le \|x_k+a_k-x^*\|^2-\frac{2\lambda_k}{1-\gamma}(f(x_k)-f(x^*))-\frac{2\gamma\lambda_k}{(1-\gamma)^2}(f(x_k)-f(x_{k-1}))\\ &\quad+\left(\frac{\lambda_k^2}{(1-\gamma)^2}-\frac{\lambda_k}{(1-\gamma)L}-\frac{\gamma\lambda_k\lambda_{k-1}}{(1-\gamma)^2}\right)\|\nabla f(x_k)\|^2\\ &\quad+\left(\frac{\gamma\lambda_k\lambda_{k-1}}{(1-\gamma)^2}-\frac{\gamma\lambda_k}{L(1-\gamma)^2}\right)\|\nabla f(x_k)-\nabla f(x_{k-1})\|^2\\ &\quad-\frac{\gamma\lambda_k\lambda_{k-1}}{(1-\gamma)^2}\|\nabla f(x_{k-1})\|^2.\end{aligned}\tag{23}$$

Consequently, for all $k\ge 1$, multiplying by $\frac{(1-\gamma)^2}{2\lambda_k}$, we get:

$$\begin{aligned}\frac{(1-\gamma)^2}{2\lambda_k}\|x_{k+1}+a_{k+1}-x^*\|^2 &\le \frac{(1-\gamma)^2}{2\lambda_k}\|x_k+a_k-x^*\|^2-(1-\gamma)(f(x_k)-f(x^*))\\ &\quad-\gamma(f(x_k)-f(x_{k-1}))-\frac{\gamma\lambda_{k-1}}{2}\|\nabla f(x_{k-1})\|^2\\ &\quad+\underbrace{\frac{1}{2}\left(\lambda_k-\frac{1-\gamma}{L}-\gamma\lambda_{k-1}\right)\|\nabla f(x_k)\|^2}_{A_k}\\ &\quad+\underbrace{\frac{1}{2}\left(\gamma\lambda_{k-1}-\frac{\gamma}{L}\right)\|\nabla f(x_k)-\nabla f(x_{k-1})\|^2}_{B_k}.\end{aligned}\tag{24}$$

Recall that since $\overline{k}$ is the smallest integer satisfying inequality (6) in Lemma 3.2, we have

$\lambda_{\overline{k}-1} = \frac{\eta_1 \|x_{\overline{k}} - x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}}) - \nabla f(x_{\overline{k}-1})\|}$. It follows that for all $k \geq \overline{k}$:

$$\begin{aligned}\lambda_k = (1+\varepsilon_{k-1})\lambda_{k-1} = \ldots = \prod_{j=\overline{k}}^{k}(1+\varepsilon_{j-1})\lambda_{\overline{k}-1} &\leq e^{\sum_{j=\overline{k}}^{k}\varepsilon_{j-1}}\lambda_{\overline{k}-1} \\ &\leq e^{\sum_{j=\overline{k}}^{+\infty}\varepsilon_{k-1}} \frac{\eta_1\|x_{\overline{k}} - x_{\overline{k}-1}\|}{\|\nabla f(x_{\overline{k}}) - \nabla f(x_{\overline{k}-1})\|} \overset{(19)}{\leq} \min\left\{\frac{1-\gamma}{L} + \gamma\lambda_{\min}, \frac{1}{L}\right\}. \end{aligned} \tag{25}$$

From (25), we derive that $A_k \leq 0$ and $B_k \leq 0$ for all $k \geq \overline{k}$. Now, taking $K > \overline{k}$ and summing (24) over $k = 1, \ldots, K$, we obtain:

$$\begin{aligned}\sum_{k=1}^{K}\frac{(1-\gamma)^2}{2\lambda_k}\|x_{k+1} + a_{k+1} - x^*\|^2 \leq \sum_{k=1}^{K}\frac{(1-\gamma)^2}{2\lambda_k}\|x_k + a_k - x^*\|^2 - (1-\gamma)\sum_{k=1}^{K}(f(x_k) - f_*) \\ - \gamma\sum_{k=1}^{K}(f(x_k) - f(x_{k-1})) + \sum_{k=1}^{\overline{k}-1}(A_k + B_k).\end{aligned}$$

Rearranging the terms yields:

$$\begin{aligned}(1-\gamma)\sum_{k=1}^{K}(f(x_k) - f_*) &\leq -\sum_{k=1}^{K-1}\frac{(1-\gamma)^2}{2\lambda_k}\|x_{k+1} + a_{k+1} - x^*\|^2 - \frac{(1-\gamma)^2}{2\lambda_K}\|x_{K+1} + a_{K+1} - x^*\|^2 \\ &\quad + \sum_{k=2}^{K}\frac{(1-\gamma)^2}{2\lambda_k}\|x_k + a_k - x^*\|^2 + \frac{(1-\gamma)^2}{2\lambda_1}\|x_1 + a_1 - x^*\|^2 \\ &\quad - \gamma(f(x_K) - f(x_0)) + \sum_{k=1}^{\overline{k}-1}(A_k + B_k) \\ &\leq -\sum_{k=1}^{K-1}\frac{(1-\gamma)^2}{2\lambda_k}\|x_{k+1} + a_{k+1} - x^*\|^2 + \sum_{k=1}^{K-1}\frac{(1-\gamma)^2}{2\lambda_{k+1}}\|x_{k+1} + a_{k+1} - x^*\|^2 \\ &\quad + \frac{(1-\gamma)^2}{2\lambda_1}\|x_1 + a_1 - x^*\|^2 - \gamma(f(x_K) - f(x_0)) + \sum_{k=1}^{\overline{k}-1}(A_k + B_k) \\ &\leq (1-\gamma)^2\sum_{k=1}^{K-1}\left(\frac{1}{2\lambda_{k+1}} - \frac{1}{2\lambda_k}\right)\|x_{k+1} + a_{k+1} - x^*\|^2 \\ &\quad + \frac{(1-\gamma)^2}{2\lambda_1}\|x_1 + a_1 - x^*\|^2 - \gamma(f(x_K) - f(x_0)) + \sum_{k=1}^{\overline{k}-1}(A_k + B_k).\end{aligned} \tag{26}$$

Recall from Lemma 3.2 that with the chosen $\overline{k}$, we have $\lambda_{k+1} > \lambda_k$ for all $k \geq \overline{k}$. Therefore:

$$(1-\gamma)^2\sum_{k=\overline{k}}^{K-1}\left(\frac{1}{2\lambda_{k+1}} - \frac{1}{2\lambda_k}\right)\|x_{k+1} + a_{k+1} - x^*\|^2 \leq 0.$$

Combining this with (26), we have:

$$(1-\gamma)\sum_{k=1}^{K}(f(x_k)-f_*)+\gamma(f(x_K)-f_*)\leq D,$$

where D is defined as:

$$D=\gamma\left(f(x_0)-f_*\right)+(1-\gamma)^2\sum_{k=1}^{\overline{k}-1}\left(\frac{1}{2\lambda_{k+1}}-\frac{1}{2\lambda_k}\right)\|x_{k+1}+a_{k+1}-x^*\|^2 + \frac{(1-\gamma)^2}{2\lambda_1}\|x_1+a_1-x^*\|^2+\sum_{k=1}^{\overline{k}-1}(A_k+B_k).$$

By the convexity of f, we conclude that

$$f(\overline{x}_K)-f_*\leq\frac{D}{\gamma+(1-\gamma)K}=O\left(\frac{1}{K}\right),\quad\forall K>\overline{k}. \tag{27}$$

where

$$\overline{x}_K=\frac{\gamma x_K+\sum_{k=1}^{K}x_k}{\gamma+(1-\gamma)K}.$$

□

Remark 3.3. We highlight the following items regarding Algorithm 2 (**NGDn**):

(i) The classical Nesterov accelerated gradient method has strong convergence rate $O(\frac{1}{k^2})$ under the same assumptions but typically requires a fixed stepsize $\lambda_k = 1/L$ or a backtracking line search procedure (see Section 2.2, Theorem 2.2.2 [4]). While the former requires knowledge of the global constant L, which is often unavailable; the latter increases computational cost per iteration. Although, the ergodic convergence rate of Algorithm 2 is $O(1/K)$ from some fixed iteration, it overcomes the limitations of classical stepsize used in classical Nesterov gradient descent method by adapting the stepsize dynamically using local Lipschitz constant of f. Its advantages will be demonstrated more clearly by numerical experiments for benchmark examples in machine learning in Section 5.

(ii) It is worth noting that the authors of [9] also proposed AdGD-accel which combines AdGD with the Nesterov's method technique. However, the convergence of AdGD-accel has not yet been provided.

4 Stochastic versions of NGDh and NGDn

In this section, we extend our proposed methods to the stochastic setting, which is essential for large-scale machine learning and deep learning applications. We propose Algorithm 3 (SNGDh) and Algorithm 4 (SNGDn) for solving Problem (1) where the objective function is potentially

nonconvex and takes the form of a finite sum problem.

$$\min_{x\in\mathbb{R}^n}\left[f(x)=\frac{1}{d}\sum_{i=1}^{d}f_i(x)\right], \tag{28}$$

This form of f is prevalent in machine learning and deep learning tasks where x corresponds to the model parameters, $f_i(x)$ represents the loss on the training point i and the aim is to minimize the average loss $f(x)$ across training points. The problem can be solved using SGD:

$$x_{k+1}=x_k-\lambda_k\nabla f_{\xi_k}(x_k),$$

where $\lambda_k>0$ is the step-size and $\nabla f_{\xi_k}(x_k)$ is an unbiased stochastic estimator of the full gradient, satisfying $\mathbb{E}(\nabla f_{\xi_k}(x_k))=\nabla f(x_k)$. In practice, this estimator is computed using a mini-batch $\xi_k\subseteq\{1,\ldots,d\}$. Crucially, these mini-batches are drawn i.i.d. from the uniform distribution over the dataset at each iteration, yielding $\nabla f_{\xi_k}(x_k)=\frac{1}{|\xi_k|}\sum\limits_{i\in\xi_k}\nabla f_i(x_k)$ to ensure the unbiasedness.

To define Algorithm 3 (SNGDh) and Algorithm 4 (SNGDn) we need the following assumption.

Assumption 4.1. *(Bounded Second Moment). There exists a constant $\sigma>0$ such that the stochastic estimator satisfies:*

$$\mathbb{E}\left(\|\nabla f_{\xi_k}(x_k)\|^2\right)\leq\sigma^2,\quad\forall k\geq 0.$$

Algorithm 3 Stochastic version of NGDh (SNGDh)

Initialization. Select $\lambda_0>0$, $0<\eta_1<\eta_0$, $0\leq\gamma<1$ and a real positive sequence $\{\varepsilon_k\}$, $x_0\in\mathbb{R}^n$, $\lambda_{\max}>\lambda_0$, $\lambda_{\max}$ is a sufficiently large number, and ξ_0 is the initial sample.
$v_1=\nabla f_{\xi_0}(x_0)$
$x_1=x_0-\lambda_0 v_1$
for $k=1,2,\ldots$ **do**
  Sampling ξ_k
  if $\|\nabla f_{\xi_k}(x_k)-\nabla f_{\xi_k}(x_{k-1})\|>\frac{\eta_0}{\lambda_{k-1}}\|x_k-x_{k-1}\|$ **then**
    $\lambda_k=\eta_1\frac{\|x_k-x_{k-1}\|}{\left\|\nabla f_{\xi_k}(x_k)-\nabla f_{\xi_k}(x_{k-1})\right\|}$
  else
    $\lambda_k=\min\{(1+\varepsilon_{k-1})\lambda_{k-1},\lambda_{\max}\}$
  end if
  $v_{k+1}=\gamma v_k+\nabla f_{\xi_k}(x_k)$
  $x_{k+1}=x_k-\lambda_k v_{k+1}$
end for

Algorithm 4 Stochastic version of NGDn (SNGDn)

Initialization. Select $\lambda_0 > 0$, $0 < \eta_1 < \eta_0$, $0 \leq \gamma < 1$ and a real positive sequence $\{\varepsilon_k\}$, $x_0 \in \mathbb{R}^n$, $\lambda_{\max} > \lambda_0$, $\lambda_{\max}$ is a sufficiently large number, and ξ_0 is the initial sample.
$v_1 = \nabla f_{\xi_0}(x_0)$
$x_1 = x_0 - \lambda_0 v_1$
for $k = 1, 2, \ldots$ **do**
    Sampling ξ_k
    if $\|\nabla f_{\xi_k}(x_k) - \nabla f_{\xi_k}(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}} \|x_k - x_{k-1}\|$ **then**
        $\lambda_k = \eta_1 \frac{\|x_k - x_{k-1}\|}{\|\nabla f_{\xi_k}(x_k) - \nabla f_{\xi_k}(x_{k-1})\|}$
    else
        $\lambda_k = \min\{(1 + \varepsilon_{k-1})\lambda_{k-1}, \lambda_{\max}\}$
    end if
    $v_{k+1} = \gamma v_k + \nabla f_{\xi_k}(x_k)$
    $x_{k+1} = x_k - \lambda_k (\gamma v_{k+1} + \nabla f_{\xi_k}(x_k))$
end for

Remark 4.1. It is worth noting that there are slight differences in the stepsize selection for Algorithms 3 and 4 compared to Algorithms 1 and 2. Specifically, we relax the condition imposed on the sequence $\{\varepsilon_k\}$ by not requiring $\sum_{k=0}^{+\infty} \varepsilon_k < +\infty$.

The bounded second moment assumption (Assumption 4.1) is very common, but it fails when the objective function f is strongly convex (as shown by Lam et al.[18]). Therefore, in this paper we will investigate the convergence of SNGDh and SNGDn without using the strongly convex assumption imposed on f. We only use the $L-$ smoothness property as follows.

Assumption 4.2. *(Smoothness) Each function $f_i : \mathbb{R}^n \to \mathbb{R}, i \in \{1, \ldots, d\}$, is L-smooth, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L \|x - y\|, \forall x, y \in \mathbb{R}^n.$$

Assumption 4.2 implies that f is also L-smooth.

Remark 4.2. It should be emphasized that there are two major distinctions between our proposed SNGD schemes and the related AdSGD algorithm given in [9]:

- **Acceleration:** SNGDh and SNGDn incorporate an explicit momentum term $\gamma \geq 0$. When $\gamma = 0$, they revert to a stochastic gradient descent without acceleration, whereas AdSGD is inherently a one-step method.
- **Nonconvexity:** The convergence analysis of AdSGD typically relies on the strong convexity of the objective function. In contrast, we establish convergence results for SNGDh and SNGDn in the nonconvex setting, which is more relevant for deep neural networks.

Next, in order to prove the convergence of our stochastic algorithms, we should prepare auxilary results regarding the boundedness of the sequence of stepsizes generated by Algorithms 3 and 4 in the following lemma.

Lemma 4.1. *Suppose that Problem* (1) *satisfies Assumptions 2.1, 4.1 and 4.2. Let* $\{\lambda_k\}$ *be the sequence generated by Algorithm 3 (or Algorithm 4, respectively). Then,*

$$\min\left\{\lambda_0, \frac{\eta_1}{L}\right\} = \lambda_{\min} \leq \lambda_k \leq \lambda_{\max}, \quad \forall k \geq 0. \tag{29}$$

Proof. It is evident that $\lambda_k \geq \min\left\{\lambda_0, \frac{\eta_1}{L}\right\}$ holds for $k = 0$. For $k \geq 1$, if $\|\nabla f_{\xi_k}(x_k) - \nabla f_{\xi_k}(x_{k-1})\| > \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\|$, then

$$\lambda_k = \frac{\eta_1\|x_k - x_{k-1}\|}{\|\nabla f_{\xi_k}(x_k) - \nabla f_{\xi_k}(x_{k-1})\|} \geq \frac{\eta_1}{L},$$

which follows from the L-smoothness of f_{ξ_k}. In the remaining case, where $\|\nabla f_{\xi_k}(x_k) - \nabla f_{\xi_k}(x_{k-1})\| \leq \frac{\eta_0}{\lambda_{k-1}}\|x_k - x_{k-1}\|$, we have

$$\lambda_k = (1 + \varepsilon_{k-1})\lambda_{k-1} \geq \lambda_{k-1}.$$

By induction, we conclude that $\lambda_k \geq \min\left\{\lambda_0, \frac{\eta_1}{L}\right\}$ for all $k \geq 0$.

Similarly, the upper bound is proved by induction. For $k = 0$, $\lambda_0 \leq \lambda_{\max}$ clearly holds. Suppose that $\lambda_i \leq \lambda_{\max}$ for all $i \leq k$. For the next step, if $\|\nabla f_{\xi_{k+1}}(x_{k+1}) - \nabla f_{\xi_{k+1}}(x_k)\| > \frac{\eta_0}{\lambda_k}\|x_{k+1} - x_k\|$, then $\frac{\eta_0\|x_{k+1}-x_k\|}{\|\nabla f_{\xi_{k+1}}(x_{k+1})-\nabla f_{\xi_{k+1}}(x_k)\|} < \lambda_k$. Consequently, since $\eta_1 < \eta_0$:

$$\lambda_{k+1} = \frac{\eta_1\|x_{k+1} - x_k\|}{\|\nabla f_{\xi_{k+1}}(x_{k+1}) - \nabla f_{\xi_{k+1}}(x_k)\|} < \frac{\eta_0\|x_{k+1} - x_k\|}{\|\nabla f_{\xi_{k+1}}(x_{k+1}) - \nabla f_{\xi_{k+1}}(x_k)\|} < \lambda_k \leq \lambda_{\max}. \tag{30}$$

In the other case, where $\|\nabla f_{\xi_{k+1}}(x_{k+1}) - \nabla f_{\xi_{k+1}}(x_k)\| \leq \frac{\eta_0}{\lambda_k}\|x_{k+1} - x_k\|$, we have:

$$\lambda_{k+1} = \min\{(1 + \varepsilon_{k-1})\,\lambda_{k-1}, \lambda_{\max}\} \leq \lambda_{\max}. \tag{31}$$

Both (30) and (31) imply $\lambda_{k+1} \leq \lambda_{\max}$. The induction proof is complete. □

The next lemma provides useful properties for the convergence analysis of SNGDh and SNGDn.

Lemma 4.2. *Suppose that Problem* (1) *satisfies Assumptions 4.1 and 4.2. Let* $\{x_k\}$ *and* $\{v_k\}$ *be defined by Algorithm 3 (or Algorithm 4, respectively). Then we have:*

(i)

$$v_{k+1} = \sum_{i=0}^{k} \gamma^i \nabla f_{\xi_{k-i}}(x_{k-i}). \tag{32}$$

(ii)

$$\mathbb{E}\left[\|v_{k+1}\|^2\right] \leq \frac{\sigma^2}{(1-\gamma)^2}, \quad \forall k \geq 0. \tag{33}$$

(iii)

$$\mathbb{E}\left[\langle \nabla f(x_k), \nabla f_{\xi_k}(x_k)\rangle\right] = \mathbb{E}\left[\|\nabla f(x_k)\|^2\right], \quad \forall k \geq 1. \tag{34}$$

Proof. (i) This follows immediately from the update rule of v_{k+1}.

(ii) Using the expansion of the squared norm and the independence assumption:

$$\begin{aligned}
\mathbb{E}\left[\|v_{k+1}\|^2\right] &= \mathbb{E}\left[\left\|\sum_{i=0}^{k} \gamma^i \nabla f_{\xi_{k-i}}(x_{k-i})\right\|^2\right] \\
&= \mathbb{E}\left[\left\langle \sum_{i=0}^{k} \gamma^i \nabla f_{\xi_{k-i}}(x_{k-i}), \sum_{j=0}^{k} \gamma^j \nabla f_{\xi_{k-j}}(x_{k-j}) \right\rangle\right] \\
&= \mathbb{E}\left[\sum_{i=0}^{k}\sum_{j=0}^{k} \gamma^i\gamma^j \langle \nabla f_{\xi_{k-i}}(x_{k-i}), \nabla f_{\xi_{k-j}}(x_{k-j})\rangle\right] \\
&\leq \mathbb{E}\left[\sum_{i=0}^{k}\sum_{j=0}^{k} \gamma^i\gamma^j \left(\frac{\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2}{2} + \frac{\|\nabla f_{\xi_{k-j}}(x_{k-j})\|^2}{2}\right)\right] \\
&\leq \sum_{i=0}^{k}\sum_{j=0}^{k} \gamma^i\gamma^j \mathbb{E}\left[\frac{\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2}{2} + \frac{\|\nabla f_{\xi_{k-j}}(x_{k-j})\|^2}{2}\right] \\
&\leq \sum_{i=0}^{k}\sum_{j=0}^{k} \gamma^i\gamma^j \sigma^2 \leq \frac{\sigma^2}{(1-\gamma)^2}.
\end{aligned}$$

(iii) Using the law of iterated expectations and the unbiasedness property:

$$\begin{aligned}
\mathbb{E}\left[\langle \nabla f(x_k), \nabla f_{\xi_k}(x_k)\rangle\right] &= \mathbb{E}\left[\mathbb{E}\left[\langle \nabla f(x_k), \nabla f_{\xi_k}(x_k)\rangle \mid x_k\right]\right] \\
&= \mathbb{E}\left[\langle \nabla f(x_k), \mathbb{E}[\nabla f_{\xi_k}(x_k) \mid x_k]\rangle\right] \\
&= \mathbb{E}\left[\|\nabla f(x_k)\|^2\right], \quad \forall k \geq 1.
\end{aligned} \tag{35}$$

□

4.1 Convergence analysis of Algorithm 3 (SNGDh)

In this section, we study the convergence of Algorithm 3. We first establish a descent-type lemma, which plays a crucial role in deriving the final convergence result.

Lemma 4.3. *Under Assumptions 4.1, 4.2, and 2.1, let $\{x_k\}, \{v_k\}, \{\lambda_k\}$ be the sequences defined by Algorithm 3. Then,*

$$\mathbb{E}\left[\langle \nabla f(x_k), v_{k+1}\rangle\right] \geq \sum_{i=0}^{k} \gamma^i \mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max} L \gamma \sigma^2}{(1-\gamma)^3}. \tag{36}$$

Proof. Consider the expansion:

$$\begin{aligned}\langle \nabla f(x_k), v_{k+1}\rangle &= \sum_{i=0}^{k} \gamma^i \langle \nabla f(x_k), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle \\ &= \sum_{i=0}^{k} \gamma^i \langle \nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle + \sum_{i=1}^{k} \gamma^i \underbrace{\langle \nabla f(x_k) - \nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle}_{H_i}.\end{aligned} \tag{37}$$

By Assumption 4.2, f is L-smooth. Thus, we have:

$$\begin{aligned}\|\nabla f(x_k) - \nabla f(x_{k-i})\|^2 &\le L^2 \|x_k - x_{k-i}\|^2 \le L^2 \left\| \sum_{l=1}^{i} (x_{k-l+1} - x_{k-l}) \right\|^2 \\ &\le L^2 \left\| \sum_{l=1}^{i} \lambda_{k-l} v_{k-l+1} \right\|^2 \le L^2 \left\| \sum_{l=1}^{i} \lambda_{\max} v_{k-l+1} \right\|^2 \\ &\le \lambda_{\max}^2 L^2 i \sum_{l=1}^{i} \|v_{k-l+1}\|^2, \quad \forall i = 1, \dots, k.\end{aligned} \tag{38}$$

On the other hand, applying Cauchy-Schwarz and Young's inequality:

$$\begin{aligned}H_i &\ge -\frac{1}{2}\left(\frac{1-\gamma}{i\lambda_{\max} L} \|\nabla f(x_k) - \nabla f(x_{k-i})\|^2 + \frac{i\lambda_{\max} L}{1-\gamma} \|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2 \right) \\ &\overset{(38)}{\ge} -\frac{\lambda_{\max} L}{2}\left((1-\gamma) \sum_{l=1}^{i} \|v_{k-l+1}\|^2 + \frac{i}{1-\gamma} \|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2 \right).\end{aligned} \tag{39}$$

Therefore, substituting back into (37), we get:

$$\begin{aligned}\langle \nabla f(x_k), v_{k+1}\rangle \ge &\sum_{i=0}^{k} \gamma^i \langle \nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle \\ &- \lambda_{\max} L \sum_{i=1}^{k} \frac{\gamma^i}{2}\left((1-\gamma) \sum_{l=1}^{i} \|v_{k-l+1}\|^2 + \frac{i}{1-\gamma} \|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2 \right).\end{aligned} \tag{40}$$

Taking the expectation on both sides of (40):

$$\begin{aligned}\mathbb{E}\left[\langle \nabla f(x_k), v_{k+1}\rangle\right] \ge &\sum_{i=0}^{k} \gamma^i \mathbb{E}\left[\langle \nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle\right] \\ &- \lambda_{\max} L \sum_{i=1}^{k} \frac{\gamma^i}{2}\left((1-\gamma) \sum_{l=1}^{i} \mathbb{E}\left[\|v_{k-l+1}\|^2\right] + \frac{i}{1-\gamma} \mathbb{E}\left[\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2\right] \right).\end{aligned} \tag{41}$$

Using Lemmas 2.2 and 4.2, combined with Assumption 4.1, we deduce:

$$\begin{aligned}\mathbb{E}\left[\langle \nabla f(x_k), v_{k+1}\rangle\right] &\geq \sum_{i=0}^{k} \gamma^i \mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] \\ &\quad - \lambda_{\max} L \sum_{i=1}^{k} \frac{\gamma^i}{2}\left((1-\gamma)\left(\sum_{l=1}^{i}\frac{\sigma^2}{(1-\gamma)^2}\right) + \frac{i}{1-\gamma}\sigma^2\right) \\ &\geq \sum_{i=0}^{k} \gamma^i \mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max}\gamma L \sigma^2}{(1-\gamma)^3}. \end{aligned} \tag{42}$$

□

The following theorem establishes the worst-case bound for the expectation of $\|\nabla f(x_k)\|^2$, implying that the best iterate from the sequence $\{x_k\}$ resides in a neighborhood of a stationary point of f.

Theorem 4.1. *Consider Problem* (1) *under Assumptions 4.1, 4.2, and 2.1. Then, the sequence* $\{x_k\}$ *generated by Algorithm 3 satisfies:*

$$\min_{k=0,\ldots,K-1} \mathbb{E}\left[\|\nabla f(x_k)\|^2\right] \leq \frac{(1-\gamma)\left(f(x_0)-f_*\right)}{\lambda_{\min}(K+\gamma^2-\gamma)} + \frac{K}{(K+\gamma^2-\gamma)}\left(\frac{\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^2} + \frac{L\lambda_{\max}^2\sigma^2}{2\lambda_{\min}(1-\gamma)}\right). \tag{43}$$

Proof. By Assumption 4.2, f is L-smooth. Therefore, from Lemma 2.1 (inequality (2.1)), we have:

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + \langle x_{k+1}-x_k, \nabla f(x_k)\rangle + \frac{L}{2}\|x_{k+1}-x_k\|^2 \\ &\leq f(x_k) - \lambda_{k+1}\langle v_{k+1}, \nabla f(x_k)\rangle + \frac{L}{2}\lambda_{k+1}^2\|v_{k+1}\|^2. \end{aligned} \tag{44}$$

Recall from Lemma 4.2 that $\lambda_{\min} \leq \lambda_k \leq \lambda_{\max} < \infty$ for all $k \geq 0$. Taking the expectation of (44):

$$\begin{aligned}\mathbb{E}\left[f(x_{k+1})\right] &\leq \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\mathbb{E}\left[\langle v_{k+1}, \nabla f(x_k)\rangle\right] + \frac{L}{2}\lambda_{\max}^2\mathbb{E}\left[\|v_{k+1}\|^2\right] \\ &\overset{(36),(33)}{\leq} \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\left(\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^3}\right) + \frac{L}{2}\lambda_{\max}^2\frac{\sigma^2}{(1-\gamma)^2} \\ &\leq \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\left(\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right]\right) + \frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^3} + \frac{L\lambda_{\max}^2\sigma^2}{2(1-\gamma)^2}. \end{aligned} \tag{45}$$

Summing (45) over $k \in \{0,\ldots,K-1\}$ and rearranging terms yields:

$$\lambda_{\min}\sum_{k=0}^{K-1}\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] \leq f(x_0) - \mathbb{E}\left[f(x_K)\right] + K\left(\frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^3} + \frac{L\lambda_{\max}^2\sigma^2}{2(1-\gamma)^2}\right). \tag{46}$$

On the other hand, changing the order of summation:

$$\begin{aligned}\sum_{k=0}^{K-1}\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] &= \sum_{k=0}^{K-1}\sum_{j=0}^{k}\gamma^{k-j}\mathbb{E}\left[\|\nabla f(x_j)\|^2\right] \\ &= \sum_{j=0}^{K-1}\mathbb{E}\left[\|\nabla f(x_j)\|^2\right]\sum_{k=j}^{K-1}\gamma^{k-j} \\ &= \frac{1}{1-\gamma}\sum_{j=0}^{K-1}\mathbb{E}\left[\|\nabla f(x_j)\|^2\right](1-\gamma^{K-j}). \end{aligned} \tag{47}$$

Observing that

$$\sum_{j=0}^{K-1}(1-\gamma^{K-j}) = K - \gamma\frac{1-\gamma^K}{1-\gamma} \geq K - \frac{\gamma}{1-\gamma}, \tag{48}$$

and combining (46), (47), with (48), we derive:

$$\lambda_{\min}\left(\frac{K}{1-\gamma}-\gamma\right)\min_{k=0,\ldots,K-1}\mathbb{E}\left[\|\nabla f(x_k)\|^2\right] \leq f(x_0)-f_* + K\left(\frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^3}+\frac{L\lambda_{\max}^2\sigma^2}{2(1-\gamma)^2}\right).$$

This is equivalent to:

$$\min_{k=0,\ldots,K-1}\mathbb{E}\left[\|\nabla f(x_k)\|^2\right] \leq \frac{(1-\gamma)\left(f(x_0)-f_*\right)}{\lambda_{\min}(K+\gamma^2-\gamma)} + \frac{K}{(K+\gamma^2-\gamma)}\left(\frac{\lambda_{\max}L\gamma\sigma^2}{(1-\gamma)^2}+\frac{L\lambda_{\max}^2\sigma^2}{2\lambda_{\min}(1-\gamma)}\right).$$

□

4.2 Convergence analysis of Algorithm 4 (SNGDn)

Similar to the previous section, we first prove a descent inequality in the following lemma.

Lemma 4.4. *Under Assumptions 4.1, 4.2, and 2.1, let $\{x_k\},\{v_k\},\{\lambda_k\}$ be defined by Algorithm 4. Then,*

$$\mathbb{E}\left[\langle\nabla f(x_k), v_{k+1}\rangle\right] \geq \sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max}\gamma L\sigma^2(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3}. \tag{49}$$

Proof. Consider the expansion:

$$\begin{aligned}\langle\nabla f(x_k), v_{k+1}\rangle &= \sum_{i=0}^{k}\gamma^i\langle\nabla f(x_k), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle \\ &= \sum_{i=0}^{k}\gamma^i\langle\nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle + \sum_{i=1}^{k}\gamma^i\underbrace{\langle\nabla f(x_k)-\nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle}_{N_i}. \end{aligned} \tag{50}$$

By Assumption 4.2, f is L-smooth. Thus:

$$\begin{aligned}
\|\nabla f(x_k) - \nabla f(x_{k-i})\|^2 &\leq L^2\|x_k - x_{k-i}\|^2 \leq L^2\left\|\sum_{l=1}^{i}(x_{k-l+1} - x_{k-l})\right\|^2 \\
&\leq L^2\left\|\sum_{l=1}^{i}\lambda_{k-l}\left(\gamma v_{k-l+1} + f_{\xi_{k-l}}(x_{k-l})\right)\right\|^2 \\
&\leq \lambda_{\max}^2 L^2\left\|\sum_{l=1}^{i}\left(\gamma v_{k-l+1} + f_{\xi_{k-l}}(x_{k-l})\right)\right\|^2 \\
&\leq 2\lambda_{\max}^2 L^2\left(\left\|\gamma\sum_{l=1}^{i} v_{k-l+1}\right\|^2 + \left\|\sum_{l=1}^{i} f_{\xi_{k-l}}(x_{k-l})\right\|^2\right) \\
&\leq 2i\lambda_{\max}^2 L^2\left(\gamma^2\sum_{l=1}^{i}\|v_{k-l+1}\|^2 + \sum_{l=1}^{i}\|f_{\xi_{k-l}}(x_{k-l})\|^2\right), \quad \forall i = 1,\ldots,k.
\end{aligned} \tag{51}$$

Applying Cauchy-Schwarz and Young's inequality for N_i:

$$\begin{aligned}
N_i &\geq -\frac{1}{2}\left(\frac{1-\gamma}{i\lambda_{\max}L}\|\nabla f(x_k) - \nabla f(x_{k-i})\|^2 + \frac{i\lambda_{\max}L}{1-\gamma}\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2\right) \\
&\overset{(51)}{\geq} -\frac{\lambda_{\max}L}{2}\left(2\gamma^2(1-\gamma)\sum_{l=1}^{i}\|v_{k-l+1}\|^2 + 2(1-\gamma)\sum_{l=1}^{i}\|f_{\xi_{k-l}}(x_{k-l})\|^2 + \frac{i}{1-\gamma}\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2\right).
\end{aligned} \tag{52}$$

Substituting back into (50):

$$\begin{aligned}
\langle\nabla f(x_k), v_{k+1}\rangle &\geq \sum_{i=0}^{k}\gamma^i\langle\nabla f(x_{k-i}), \nabla f_{\xi_{k-i}}(x_{k-i})\rangle \\
&\quad - \lambda_{\max}L\sum_{i=1}^{k}\frac{\gamma^i}{2}\left(2\gamma^2(1-\gamma)\sum_{l=1}^{i}\|v_{k-l+1}\|^2 + 2(1-\gamma)\sum_{l=1}^{i}\|f_{\xi_{k-l}}(x_{k-l})\|^2\right. \\
&\quad \left. + \frac{i}{1-\gamma}\|\nabla f_{\xi_{k-i}}(x_{k-i})\|^2\right).
\end{aligned} \tag{53}$$

Taking expectations and applying Lemmas 2.2, 4.2, and Assumption 4.1:

$$\begin{aligned}
\mathbb{E}\left[\langle\nabla f(x_k), v_{k+1}\rangle\right] &\geq \sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] \\
&\quad - \lambda_{\max}L\sum_{i=1}^{k}\frac{\gamma^i i}{2}\left(2\gamma^2(1-\gamma)\frac{\sigma^2}{(1-\gamma)^2} + 2(1-\gamma)\sigma^2 + \frac{1}{1-\gamma}\sigma^2\right) \\
&\geq \sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max}\gamma L\sigma^2(4\gamma^2 - 4\gamma + 3)}{2(1-\gamma)^3}.
\end{aligned} \tag{54}$$

□

The final result of this study establishes the convergence of Algorithm 4, demonstrating that the best iterate approaches a neighborhood of some stationary point of the objective function f.

Theorem 4.2. *Consider Problem* (1) *under Assumptions 4.1, 4.2, and 2.1. Then, the sequence* $\{x_k\}$ *generated by Algorithm 4 satisfies:*

$$\min_{k=0,\dots,K-1} \mathbb{E}\left[\|\nabla f(x_k)\|^2\right] \leq \frac{(1-\gamma)\left(f(x_0)-f_*\right)}{\lambda_{\min}(K+(K-\gamma)(1-\gamma))} + \\ + \frac{KL\lambda_{\max}\sigma^2}{K+(K-\gamma)(1-\gamma)}\left(\frac{\gamma(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3}+\frac{\lambda_{\max}(\gamma^2-\gamma+1)}{\lambda_{\min}(1-\gamma)^2}\right). \tag{55}$$

Proof. By Assumption 4.2, f is L-smooth. From Lemma 2.1 (inequality (2.1)):

$$f(x_{k+1}) \leq f(x_k) + \langle x_{k+1}-x_k, \nabla f(x_k)\rangle + \frac{L}{2}\|x_{k+1}-x_k\|^2 \\ \leq f(x_k) - \lambda_{k+1}\langle \gamma v_{k+1} + \nabla f_{\xi_k}(x_k), \nabla f(x_k)\rangle + \frac{L}{2}\lambda_{k+1}^2\|\gamma v_{k+1} + \nabla f_{\xi_k}(x_k)\|^2. \tag{56}$$

Using Lemma 4.2 ($\lambda_{\min} \leq \lambda_k \leq \lambda_{\max}$), we take expectations:

$$\mathbb{E}\left[f(x_{k+1})\right] \leq \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\big(\gamma\mathbb{E}\left[\langle v_{k+1}, \nabla f(x_k)\rangle\right] + \mathbb{E}\left[\langle \nabla f_{\xi_k}(x_k), \nabla f(x_k)\rangle\right]\big) \\ + L\lambda_{\max}^2\left(\gamma\mathbb{E}\left[\|v_{k+1}\|^2\right] + \mathbb{E}\left[\|\nabla f_{\xi_k}(x_k)\|^2\right]\right). \tag{57}$$

Combining with (36) and (33), we evaluate:

$$\mathbb{E}\left[f(x_{k+1})\right] \leq \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\left(\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] - \frac{\lambda_{\max}\gamma L\sigma^2(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3} + \mathbb{E}\left[\|\nabla f(x_k)\|^2\right]\right) \\ + L\lambda_{\max}^2\left(\frac{\gamma\sigma^2}{(1-\gamma)^2}+\sigma^2\right) \\ \leq \mathbb{E}\left[f(x_k)\right] - \lambda_{\min}\left(\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] + \mathbb{E}\left[\|\nabla f(x_k)\|^2\right]\right) \\ + \frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3} + \frac{L\lambda_{\max}^2\sigma^2(\gamma^2-\gamma+1)}{(1-\gamma)^2}. \tag{58}$$

Summing (58) over $k \in \{0,\dots,K-1\}$ and reformulating:

$$\lambda_{\min}\sum_{k=0}^{K-1}\left(\sum_{i=0}^{k}\gamma^i\mathbb{E}\left[\|\nabla f(x_{k-i})\|^2\right] + \mathbb{E}\left[\|\nabla f(x_k)\|^2\right]\right) \\ \leq f(x_0) - \mathbb{E}\left[f(x_K)\right] + K\left(\frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3} + \frac{L\lambda_{\max}^2\sigma^2(\gamma^2-\gamma+1)}{(1-\gamma)^2}\right). \tag{59}$$

Using the identity from (47) and (48), we substitute back to derive:

$$\lambda_{\min}\left(\frac{K}{1-\gamma}-\gamma+K\right)\min_{k=0,\dots,K-1}\mathbb{E}\left[\|\nabla f(x_k)\|^2\right]$$
$$\leq f(x_0)-f_*+K\left(\frac{\lambda_{\min}\lambda_{\max}L\gamma\sigma^2(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3}+\frac{L\lambda_{\max}^2\sigma^2(\gamma^2-\gamma+1)}{(1-\gamma)^2}\right).$$

This is equivalent to the stated result:

$$\min_{k=0,\dots,K-1}\mathbb{E}\left[\|\nabla f(x_k)\|^2\right]\leq\frac{(1-\gamma)\left(f(x_0)-f_*\right)}{\lambda_{\min}(K+(K-\gamma)(1-\gamma))}+$$
$$+\frac{KL\lambda_{\max}\sigma^2}{K+(K-\gamma)(1-\gamma)}\left(\frac{\gamma(4\gamma^2-4\gamma+3)}{2(1-\gamma)^3}+\frac{\lambda_{\max}(\gamma^2-\gamma+1)}{\lambda_{\min}(1-\gamma)^2}\right).$$

□

5 Numerical experiments

In this section, we evaluate the performance of the proposed algorithms on logistic regression problems in machine learning and neural network training in deep learning. All algorithms were implemented in Python. Our source code is available at https://github.com/hoaiphamthi/Accelerated-and-Stochastic-NGD.

5.1 Logistic Regression

In this subsection, experiments are conducted on the logistic regression problem, which minimizes the following objective function:

$$f(x)=\frac{1}{d}\sum_{i=1}^{d}\log(1+\exp(-b_i a_i^T x))+\frac{\ell}{2}\|x\|^2,$$

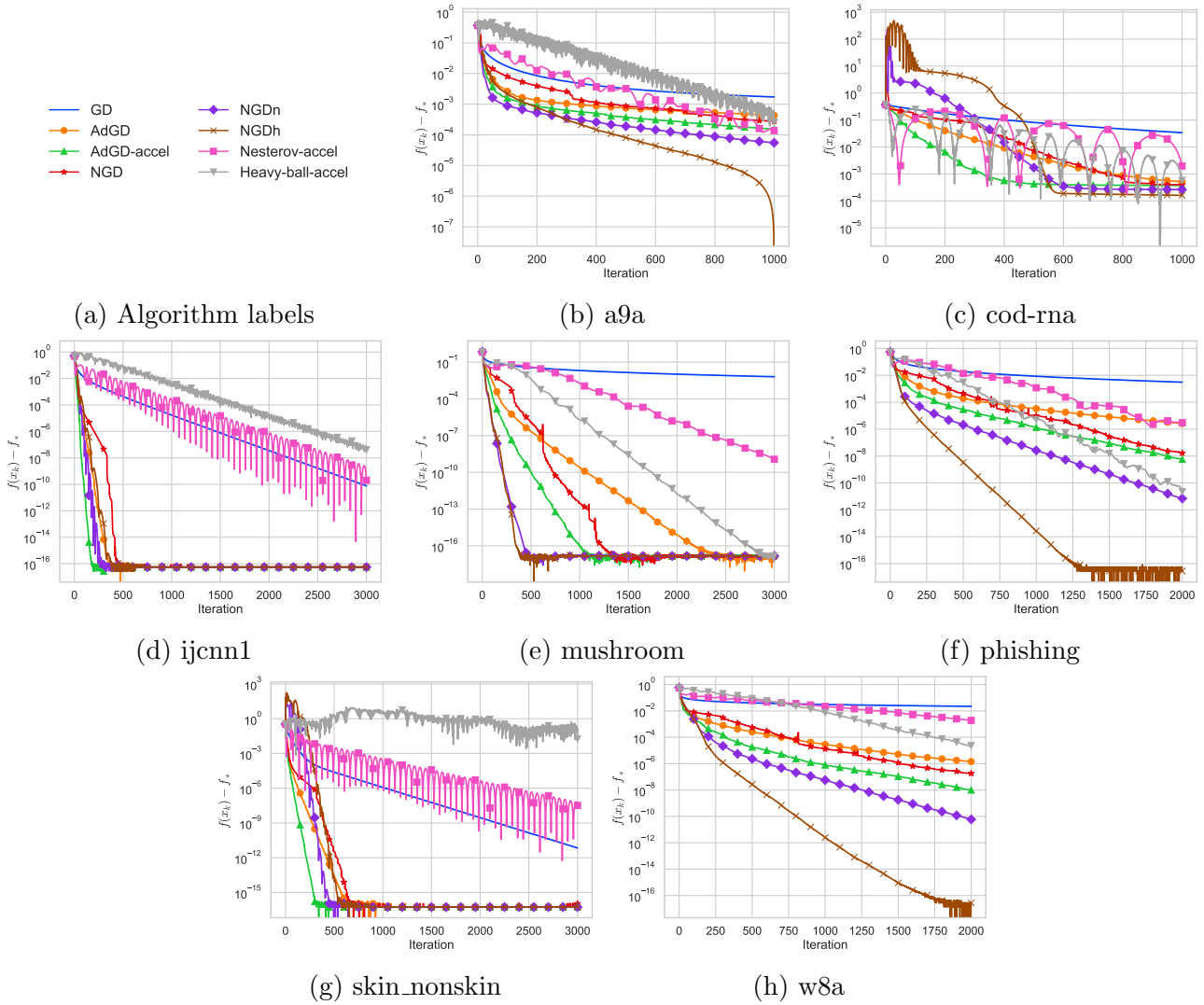
where $(a_i,b_i)\in\mathbb{R}^n\times\mathbb{R}$ for $i=1,\dots,d$ denote the observations, and $\ell>0$ is the regularization parameter, typically chosen as $\frac{1}{d}$. We compare our proposed algorithms, **NGDh** and **NGDn**, with related methods including **GD** [8], **Nesterov-accel**, **Heavy-ball-accel**, **AdGD**, and **AdGD-accel** [9]. We utilize popular benchmark datasets [1] such as `a9a`, `cod-rna`, `ijcnn1`, `mushroom`, `phishing`, `skin_nonskin` and `w8a`. In the experiments, the parameters are configured as follows:

- **GD:** $\lambda_k=\frac{1}{L},\forall k\geq 0$, where L is the smoothness constant of f.
- **Nesterov-accel:** Nesterov's accelerated gradient descent configured for strongly convex functions. We use a constant stepsize $\lambda=\frac{1}{L}$ and a fixed momentum coefficient $\gamma=\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}$.

[1] All datasets are available at https://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/

- **Heavy-ball-accel:** Polyak's Heavy Ball method using the theoretical optimal parameters for strongly convex quadratic functions. The stepsize is set to $\lambda = \frac{4}{(\sqrt{L}+\sqrt{\mu})^2}$ and the momentum coefficient is $\gamma = \left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^2$.
- **AdGD** and **AdGD-accel:** Default parameters are used as provided in the public source code[2].
- **NGD:** $\lambda_0 = 10^{-3}$, $\eta_0 = 0.2$, $\eta_1 = 0.15$, and $\varepsilon_k = \dfrac{2(\log k)^{4.5}}{k^{1.1}}$.
- **NGDn** and **NGDh:** $\lambda_0 = 0.01$, $\eta_0 = 0.2$, $\eta_1 = 0.19$, and $\varepsilon_k = \dfrac{3}{k^{1.1}}$.

As illustrated in Figure 1, both NGDh and NGDn demonstrate superior performance across most tested datasets. The results also show that on certain datasets, with a fixed stepsize, Heavy-ball-accel and Nesterov-accel algorithms exhibit severe oscillations and instability.



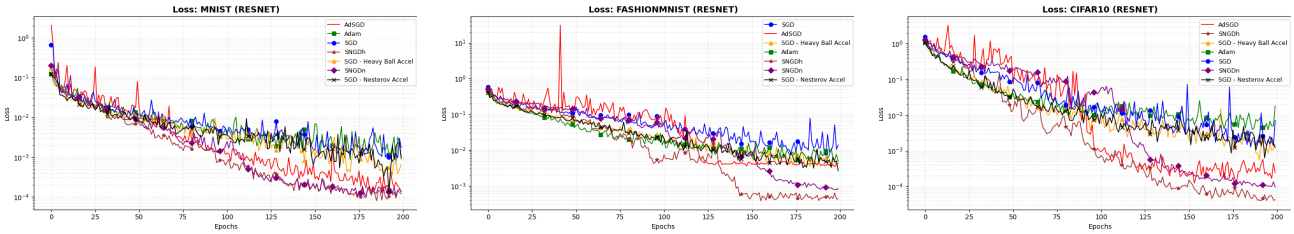
(a) Algorithm labels (b) a9a (c) cod-rna

(d) ijcnn1 (e) mushroom (f) phishing

(g) skin_nonskin (h) w8a

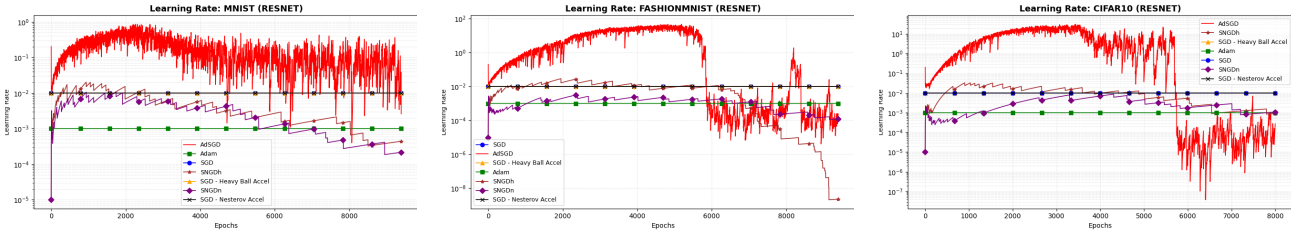
Fig. 1: The logistic regression objective results

[2]https://github.com/ymalitsky/adaptive_gd



(a) MNIST (b) FashionMNIST (c) CIFAR-10

Fig. 2: Training losses for different datasets



(a) MNIST (b) FashionMNIST (c) CIFAR-10

Fig. 3: Stepsizes for different datasets

5.2 Deep Learning Tasks

The subsequent experiments evaluate the performance of the stochastic algorithms, **SNGDh** and **SNGDn**, on the benchmark datasets **MNIST** [19], **FashionMNIST** [20], and **CIFAR-10** [21]. The compared algorithms include **SGD**, **SGD Heavy Ball Accel**, **SGD Nesterov Accel AdSGD** [9], **Adam**, **SNGDn**, and **SNGDh**. The hyperparameters are set as follows:

- **SGD**, **SGD Heavy Ball Accel** and **SGD Nesterov Accel**: Default parameters with $\lambda_k = 0.01, \forall k \geq 0$. The momentum parameter for SGD Accel is set to $\gamma = 0.9$.
- **AdSGD:** Default parameters are used as in the public source code of [9].
- **SNGDn** and **SNGDh:** $\lambda_0 = 10^{-5}$, $\eta_0 = 0.2$, $\eta_1 = 0.15$, $\varepsilon_k = \dfrac{1}{k^{0.9}}$, and $\gamma = 0.9$. The parameter $\lambda_{\max}$ is initially set to a sufficiently large positive number (e.g., $\lambda_{\max} = 10$). However, empirical observations revealed that the effective stepsizes for SNGDh and SNGDn remained very small (less than 1, see Figure 3). Consequently, we omitted this parameter in our public Python implementation.

The numerical results are presented in Figures 2 and 3. As shown in Figure 2, SNGDn and SNGDh exhibit the best performance across all datasets. Particularly on FashionMNIST and CIFAR-10, SNGDh yields the lowest training losses.

Observing the stepsize Figure 3, it is evident that our algorithms (SNGDh and SNGDn) exhibit adaptive behavior throughout the training phase. They maintain large stepsizes during the initial epochs and gradually decay upon converging to a stationary point. Consequently, our algorithms provide the best performances among the compared methods regarding objective function values.

6 Conclusions

In this paper, we have developed efficient accelerated and stochastic variants of the NGD algorithm originally proposed in [10]. By incorporating Polyak's Heavy Ball and Nesterov's momentum techniques, we address the limitations of fixed-stepsize and backtracking linesearch strategies. From some finite iteration, we establish the ergodic convergence of the proposed deterministic algorithms with a sublinear rate of $O(1/K)$ for convex objective functions, relying solely on local Lipschitz estimates rather than a global constant. Furthermore, we extend these methods to the stochastic setting to handle large-scale optimization problems. We provide a rigorous convergence analysis for the stochastic algorithms (SNGDh and SNGDn) within a nonconvex framework, proving that the expected gradient norm remains bounded under standard variance assumptions. Numerical experiments on logistic regression and deep neural network training (using MNIST, FashionMNIST, and CIFAR10 datasets) confirm that our proposed algorithms achieve superior performance and faster convergence compared to existing adaptive methods.

Acknowledgments

The authors wish to thank the editors and the anonymous referees very much for their useful comments which helped them to improve the paper greatly. The second author expresses her gratitude to Professor Nguyen Dong Yen for providing valuable materials and constructive comments to improve the quality of this paper.

References

[1] Changho Suh. *Convex Optimization for Machine Learning*. Boston-Delft: Now Publishers, 2022.

[2] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering, 2006.

[3] Boris Teodorovich Polyak. *Introduction to optimization*. New York, Optimization Software, 1987.

[4] Yurii Nesterov. *Lectures on Convex Optimization*. Springer Publishing Company, Incorporated, 2 edition, 2018.

[5] Amir Beck. *First order methods in optimization*. Society for Industrial and Applied Mathematics, USA, 2017.

[6] Amir Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*. Society for Industrial and Applied Mathematics, 2014.

[7] Dimitri Panteli Bertsekas. *Nonlinear programming*. Athena Scientific, 3 edition, 2016.

[8] Claude Lemaréchal. Cauchy and the gradient method. *Doc. Math. Extra Vol. Optimization Stories*, pages 251–254, 2012.

[9] Yura Malitsky and Konstantin Mishchenko. Adaptive gradient descent without descent. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6702–6712. PMLR, 7 2020.

[10] Pham Thi Hoai, Nguyen The Vinh, and Nguyen Phung Hai Chung. A novel stepsize for gradient descent method. *Operations Research Letters*, 53:107072, 2024.

[11] Boris Teodorovich Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[12] Euhanna Ghadimi, Hamid Reza Feyzmahdavian, and Mikael Johansson. Global convergence of the heavy-ball method for convex optimization. In *2015 European Control Conference (ECC)*, pages 310–315, 2015.

[13] Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60, 2018.

[14] Arkadii S. Nemirovski, Anatoli B. Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[15] Benjamin Recht, Christopher Re, Stephen Wright, and Feng Niu. Hogwild!: A lock-free approach to parallelizing stochastic gradient descent. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

[16] Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: Optimal algorithms for stochastic strongly-convex optimization. *Journal of Machine Learning Research*, 15(71):2489–2512, 2014.

[17] Alexandre D'efossez, Léon Bottou, Francis R. Bach, and Nicolas Usunier. A simple convergence proof of adam and adagrad. *Transactions on Machine Learning Research*, 2022.

[18] Lam Nguyen, Phuong Ha Nguyen, Marten van Dijk, Peter Richtarik, Katya Scheinberg, and Martin Takac. SGD and hogwild! Convergence without the bounded gradients assumption. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3750–3758. PMLR, 7 2018.

[19] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[20] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747 [cs.LG]*, 2017.

[21] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.