Restarted Nonconvex Accelerated Gradient Descent: No More Polylogarithmic Factor in the $\mathcal{O}(\epsilon^{-7/4})$ Complexity

Huan Li [1] **Zhouchen Lin** [2 3 4]

Abstract

This paper studies the accelerated gradient descent for general nonconvex problems under the gradient Lipschitz and Hessian Lipschitz assumptions. We establish that a simple restarted accelerated gradient descent (AGD) finds an ϵ-approximate first-order stationary point in $\mathcal{O}(\epsilon^{-7/4})$ gradient computations with simple proofs. Our complexity does not hide any polylogarithmic factors, and thus it improves over the state-of-the-art one by the $\mathcal{O}(\log \frac{1}{\epsilon})$ factor. Our simple algorithm only consists of Nesterov's classical AGD and a restart mechanism, and it does not need the negative curvature exploitation or the optimization of regularized surrogate functions. Technically, our simple proof does not invoke the analysis for the strongly convex AGD, which is crucial to remove the $\mathcal{O}(\log \frac{1}{\epsilon})$ factor.

1. Introduction

Nonconvex optimization has emerged increasingly popular in machine learning and a lot of machine learning tasks can be formulated as nonconvex problems, such as deep learning (LeCun et al., 2015). This paper considers the following general nonconvex problem:

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

where $f(\mathbf{x})$ is bounded from below and has Lipschitz continuous gradient and Hessian.

Gradient descent, a simple and fundamental algorithm, is known to find an ϵ-approximate first-order stationary point of problem (1) (where $\|\nabla f(\mathbf{x})\| \le \epsilon$) in $\mathcal{O}(\epsilon^{-2})$ iterations (Nesterov, 2004). This rate is optimal among the first-order methods under the gradient Lipschitz condition (Cartis et al., 2010; Carmon et al., 2020). When additional structure is assumed, such as the Hessian Lipschitz condition, improvement is possible.

For convex problems, gradient descent is known to be suboptimal. In a series of celebrated works (Nesterov, 1983; 1988; 2005), Nesterov proposed several accelerated gradient descent (AGD) methods, which find an ϵ-optimal solution in $\mathcal{O}(\sqrt{\frac{L}{\epsilon}})$ and $\mathcal{O}(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon})$ iterations for L-smooth general convex problems and μ-strongly convex problems, respectively, while gradient descent takes $\mathcal{O}(\frac{L}{\epsilon})$ and $\mathcal{O}(\frac{L}{\mu}\log\frac{1}{\epsilon})$ steps. Motivated by the practical superiority and rich theory of accelerated methods for convex optimization, nonconvex AGD has attracted tremendous attentions in recent years. In this paper, we aim to give a slightly faster convergence rate than the state-of-the-art one by simple proofs for a simple nonconvex AGD.

1.1. Literature Review

Nonconvex AGD has been a hot topic in the last decade. Ghadimi & Lan (2016); Li & Lin (2015); Li et al. (2017) studied the nonconvex AGD under the gradient Lipschitz condition. The efficiency is verified empirically and there is no speed improvement in theory. Carmon et al. (2017) proposed a "convex until guilty" mechanism with nested-loop under both the gradient Lipschitz and Hessian Lipschitz conditions, which finds an ϵ-approximate first-order stationary point in $\mathcal{O}(\epsilon^{-7/4}\log\frac{1}{\epsilon})$ gradient and function evaluations. Their method alternates between the minimization of a regularized surrogate function and the negative curvature exploitation, where in the former subroutine, a proximal term is added to reduce the nonconvex subproblem to a convex one.

Most literatures focus on the second-order stationary point when studying nonconvex AGD. Carmon et al. (2018) combined the regularized accelerated gradient descent and the Lanczos method, where the latter is used to search the negative curvature. Agarwal et al. (2017) implemented the cubic-regularized Newton steps carefully by using accelerated

[1]Institute of Robotics and Automatic Information Systems, College of Artificial Intelligence, Nankai University, Tianjin, China. [2]Key Laboratory of Machine Perception, School of Artificial Intelligence, Peking University, Beijing, China [3]Institute for Artificial Intelligence, Peking University [4]Pazhou Lab, Guangzhou, China. Correspondence to: <lihuanss@nankai.edu.cn, zlin@pku.edu.cn>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

method for fast approximate matrix inversion, while Carmon & Duchi (2020; 2018) employed the Krylov subspace method to approximate the cubic-regularized Newton steps. The above methods find an ϵ-approximate second-order stationary point in $\mathcal{O}(\epsilon^{-7/4}\log\frac{1}{\epsilon})$ gradient evaluations or Hessian-vector products. To avoid the Hessian-vector products, Xu et al. (2018) and Allen-Zhu & Li (2018) proposed the NEON and NEON2 first-order procedures to extract negative curvature of the Hessian, respectively. Other typical methods include the Newton-CG (Royer et al., 2020) and the second-order line-search method (Royer & Wright, 2018), which are beyond the AGD class.

The methods in (Carmon et al., 2017; 2018; Agarwal et al., 2017; Carmon & Duchi, 2020) are nested-loop algorithms. They either alternate between the negative curvature exploitation and the optimization of a regularized surrogate function using convex AGD (Carmon et al., 2018; 2017), or call the accelerated methods to solve a series of cubic regularized Newton steps (Agarwal et al., 2017; Carmon & Duchi, 2020). Jin et al. (2018) proposed the first single-loop accelerated method, which finds an ϵ-approximate second-order stationary point in $\mathcal{O}(\epsilon^{-7/4}\log\frac{1}{\epsilon})$ gradient and function evaluations. The method in (Jin et al., 2018) runs the classical AGD until some condition triggers, then calls the negative curvature exploitation, and continues on the classical AGD. It is, as far as we know, the simplest algorithm among the nonconvex accelerated methods with fast rate guarantees.

Although achieving second-order stationary point ensures the method not to get stuck at the saddle points, some researchers show that gradient descent and its accelerated variants that converge to first-order stationary point always converge to local minimum. Lee et al. (2016) established that gradient descent converges to a local minimizer almost surely with random initialization. O'Neill & Wright (2019) proved that accelerated method is unlikely to converge to strict saddle points, and diverges from the strict saddle point more rapidly than the steepest-descent method for specific quadratic objectives.

As for the lower bound, Carmon et al. (2021) established that no deterministic first-order method can find ϵ-approximate first-order stationary point of functions with Lipschitz continuous gradient and Hessian in less than $\mathcal{O}(\epsilon^{-12/7})$ gradient evaluations. There exists a gap of $\mathcal{O}(\epsilon^{-1/28}\log\frac{1}{\epsilon})$ between the lower bound and the state-of-the-art upper bound (Carmon et al., 2017; Jin et al., 2018). It remains an open problem of how to close this gap.

1.2. Contribution

All of the above methods (Carmon et al., 2017; 2018; Agarwal et al., 2017; Carmon & Duchi, 2020; Jin et al., 2018) share the $\mathcal{O}(\epsilon^{-7/4}\log\frac{1}{\epsilon})$ complexity, which has a $\mathcal{O}(\log\frac{1}{\epsilon})$ factor. To the best of our knowledge, even applying the methods designed to find second-order stationary point to the easier problem of finding first-order stationary point, the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor still cannot be removed. On the other hand, almost all the existing methods are complex with nested loops. Even the single-loop method proposed in (Jin et al., 2018) needs the negative curvature exploitation procedure.

Algorithm 1 Restarted AGD

Initialize $\mathbf{x}^{-1}=\mathbf{x}^0=\mathbf{x}_{int}$, $k=0$.
while $k<K$ **do**
  $\mathbf{y}^k=\mathbf{x}^k+(1-\theta)(\mathbf{x}^k-\mathbf{x}^{k-1})$
  $\mathbf{x}^{k+1}=\mathbf{y}^k-\eta\nabla f(\mathbf{y}^k)$
  $k=k+1$
  if $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2>B^2$ **then**
    $\mathbf{x}^{-1}=\mathbf{x}^0=\mathbf{x}^k$, $k=0$
  end if
end while
$K_0=\operatorname{argmin}_{\lfloor\frac{K}{2}\rfloor\le k\le K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$
Output $\hat{\mathbf{y}}=\frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{y}^k$

In this paper, we propose a simple restarted AGD, which has the following three advantages:

1. Our method finds an ϵ-approximate first-order stationary point in $\mathcal{O}(\epsilon^{-7/4})$ gradient computations. Our complexity does not hide any polylogarithmic factors, and thus it improves over the state-of-the-art one by the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor.

2. Our method is simple in the sense that it only consists of Nesterov's classical AGD and a restart mechanism, and it does not need the negative curvature exploitation or the optimization of regularized surrogate functions.

3. Technically, our proof is much simpler than all those in the existing literatures. Especially, we do not invoke the analysis for the strongly convex AGD, which is crucial to remove the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor.

2. Restarted Accelerated Gradient Descent

We make the following standard assumptions in this paper, where we denote $\|\cdot\|$ to be the Euclidean norm for vectors and the spectral norm for matrices uniformly.

Assumption 2.1. 1. $f(\mathbf{x})$ is L-gradient Lipschitz: $\|\nabla f(\mathbf{x})-\nabla f(\mathbf{y})\|\le L\|\mathbf{x}-\mathbf{y}\|$.

2. $f(\mathbf{x})$ is ρ-Hessian Lipschitz: $\|\nabla^2 f(\mathbf{x})-\nabla^2 f(\mathbf{y})\|\le\rho\|\mathbf{x}-\mathbf{y}\|$.

Our method is described in Algorithm 1. It runs Nesterov's classical AGD until the "if condition" triggers. Then we restart by setting $\mathbf{x}^0$ and $\mathbf{x}^{-1}$ equal to $\mathbf{x}^k$ and do the next round of AGD. The algorithm terminates when the "if condition" does not trigger in K iterations. The restart trick is motivated by (Fang et al., 2019), who proposed a ball-mechanism as the stopping criteria to analyze SGD. In contrast with other nonconvex accelerated methods, our method does not invoke any additional techniques, such as the negative curvature exploitation, the optimization of regularized surrogate functions, or the minimization of cubic Newton steps.

We present our main result in Theorem 2.2, which establishes the $\mathcal{O}(\epsilon^{-7/4})$ complexity to find an ϵ-approximate first-order stationary point.

Theorem 2.2. *Suppose that Assumption 2.1 holds. Let $\eta = \frac{1}{4L}$, $B = \sqrt{\frac{\epsilon}{\rho}}$, $\theta = 4\left(\epsilon\rho\eta^2\right)^{1/4} < 1$, $K = \frac{1}{\theta}$. Then Algorithm 1 terminates in at most $\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}$ gradient computations and the output satisfies $\|\nabla f(\hat{\mathbf{y}})\| \leq 82\epsilon$, where $\triangle_f = f(\mathbf{x}_{int}) - \min_{\mathbf{x}} f(\mathbf{x})$.*

Among the existing methods, Carmon et al. (2017) established the $\mathcal{O}\left(\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}\log\frac{L\triangle_f}{\epsilon}\right)$ complexity to find an ϵ-approximate first-order stationary point, which has the additional $\mathcal{O}(\log\frac{1}{\epsilon})$ factor compared with our one. The complexity given in other literatures concentrating on second-order stationary point, such as (Carmon et al., 2018; Agarwal et al., 2017; Carmon & Duchi, 2020; Jin et al., 2018), also has the additional $\mathcal{O}(\log\frac{1}{\epsilon})$ factor even for finding first-order stationary point. Take (Jin et al., 2018) as the example. Their Lemma 7 studies the first-order stationary point. Their proof in Lemmas 9 and 17 is built upon the analysis for strongly convex AGD, which generally needs $\mathcal{O}(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon})$ iterations such that the gradient norm will be less than ϵ, and thus the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor appears.

3. Proof of Theorem 2.2

Define $\mathcal{K}$ to be the iteration number when the "if condition" triggers in Algorithm 1, that is,

$$\mathcal{K} = \min_k \left\{ k \middle| k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2 \right\}.$$

Denote the iterations from $k=0$ to $k=\mathcal{K}$ to be one epoch. Then for each epoch except the last one, we have $1 \leq \mathcal{K} \leq K$,

$$\mathcal{K}\sum_{t=0}^{\mathcal{K}-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2, \tag{2a}$$

$$\|\mathbf{x}^k-\mathbf{x}^0\|^2 \leq k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 \leq B^2, \forall k<\mathcal{K}, \tag{2b}$$

$$\|\mathbf{y}^k-\mathbf{x}^0\| \leq \|\mathbf{x}^k-\mathbf{x}^0\|+\|\mathbf{x}^k-\mathbf{x}^{k-1}\| \leq 2B, \forall k<\mathcal{K}. \tag{2c}$$

For the last epoch, that is, the "if condition" does not trigger and the while loop breaks until $k=K$, we have

$$\|\mathbf{x}^k-\mathbf{x}^0\|^2 \leq k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 \leq B^2, \forall k \leq K, \tag{3a}$$

$$\|\mathbf{y}^k-\mathbf{x}^0\| \leq 2B, \forall k \leq K. \tag{3b}$$

We will show in Sections 3.1 and 3.2 that the function value decreases with a magnitude at least $\mathcal{O}(\epsilon^{1.5})$ in each epoch except the last one. Thus Algorithm 1 terminates in at most $\mathcal{O}(\epsilon^{-1.5})$ epochs, and accordingly $\mathcal{O}(\epsilon^{-1.75})$ total gradient computations since each epoch needs at most $\mathcal{O}(\epsilon^{-0.25})$ iterations. In the last epoch, we will show that the gradient norm at the output iterate is less than $\mathcal{O}(\epsilon)$, which is detailed in Section 3.3.

3.1. Large Gradient of $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\|$

We first consider the case when $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\|$ is large.

Lemma 3.1. *Suppose that Assumption 2.1 holds. Let $\eta \leq \frac{1}{4L}$ and $0 \leq \theta \leq 1$. When the "if condition" triggers and $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| > \frac{B}{\eta}$, then for Algorithm 1 we have*

$$f(\mathbf{x}^{\mathcal{K}}) - f(\mathbf{x}^0) \leq -\frac{B^2}{4\eta}.$$

Proof. From the L-gradient Lipschitz condition, we have

$$\begin{aligned}
&f(\mathbf{x}^{k+1})\\
\leq& f(\mathbf{y}^k) + \left\langle \nabla f(\mathbf{y}^k), \mathbf{x}^{k+1}-\mathbf{y}^k \right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|^2\\
=& f(\mathbf{y}^k) - \eta\|\nabla f(\mathbf{y}^k)\|^2 + \frac{L\eta^2}{2}\|\nabla f(\mathbf{y}^k)\|^2\\
\leq& f(\mathbf{y}^k) - \frac{7\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2,
\end{aligned} \tag{4}$$

where we use $\eta \leq \frac{1}{4L}$. From the L-gradient Lipschitz, we also have

$$f(\mathbf{x}^k) \geq f(\mathbf{y}^k) + \left\langle \nabla f(\mathbf{y}^k), \mathbf{x}^k-\mathbf{y}^k \right\rangle - \frac{L}{2}\|\mathbf{x}^k-\mathbf{y}^k\|^2.$$

So we have

$$\begin{aligned}
&f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k)\\
\leq& -\left\langle \nabla f(\mathbf{y}^k), \mathbf{x}^k-\mathbf{y}^k \right\rangle + \frac{L}{2}\|\mathbf{x}^k-\mathbf{y}^k\|^2 - \frac{7\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2\\
=& \frac{1}{\eta}\left\langle \mathbf{x}^{k+1}-\mathbf{y}^k, \mathbf{x}^k-\mathbf{y}^k \right\rangle + \frac{L}{2}\|\mathbf{x}^k-\mathbf{y}^k\|^2 - \frac{7\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2\\
=& \frac{1}{2\eta}\left(\|\mathbf{x}^{k+1}-\mathbf{y}^k\|^2 + \|\mathbf{x}^k-\mathbf{y}^k\|^2 - \|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\right)\\
&+ \frac{L}{2}\|\mathbf{x}^k-\mathbf{y}^k\|^2 - \frac{7\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2\\
\overset{a}{\leq}& \frac{5}{8\eta}\|\mathbf{x}^k-\mathbf{y}^k\|^2 - \frac{1}{2\eta}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2 - \frac{3\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2\\
\overset{b}{\leq}& \frac{5}{8\eta}\|\mathbf{x}^k-\mathbf{x}^{k-1}\|^2 - \frac{1}{2\eta}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2 - \frac{3\eta}{8}\|\nabla f(\mathbf{y}^k)\|^2,
\end{aligned}$$

where we use $L \leq \frac{1}{4\eta}$ in $\overset{a}{\leq}$ and $\|\mathbf{x}^k - \mathbf{y}^k\| = (1-\theta)\|\mathbf{x}^k - \mathbf{x}^{k-1}\| \leq \|\mathbf{x}^k - \mathbf{x}^{k-1}\|$ in $\overset{b}{\leq}$. Summing over $k = 0, \cdots, \mathcal{K}-1$ and using $\mathbf{x}^0 = \mathbf{x}^{-1}$, we have

$$\begin{aligned}
&f(\mathbf{x}^{\mathcal{K}}) - f(\mathbf{x}^0)\\
\leq&\frac{1}{8\eta}\sum_{k=0}^{\mathcal{K}-2}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2 - \frac{3\eta}{8}\sum_{k=0}^{\mathcal{K}-1}\|\nabla f(\mathbf{y}^k)\|^2\\
\overset{c}{\leq}&\frac{B^2}{8\eta} - \frac{3\eta}{8}\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\|^2 \overset{d}{\leq} \frac{B^2}{8\eta} - \frac{3B^2}{8\eta} \leq -\frac{B^2}{4\eta},
\end{aligned}$$

where we use (2b) in $\overset{c}{\leq}$ and $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| > \frac{B}{\eta}$ in $\overset{d}{\leq}$. □

3.2. Small Gradient of $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\|$

If $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| \leq \frac{B}{\eta}$, then from (2c) we have

$$\|\mathbf{x}^{\mathcal{K}} - \mathbf{x}^0\| \leq \|\mathbf{y}^{\mathcal{K}-1} - \mathbf{x}^0\| + \eta\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| \leq 3B.$$

For each epoch, denote $\mathbf{H} = \nabla^2 f(\mathbf{x}^0)$ and $\mathbf{H} = \mathbf{U}\Lambda\mathbf{U}^T$ to be its eigenvalue decomposition with $\mathbf{U}, \Lambda \in \mathbb{R}^{d\times d}$. Let λ_j be the jth eigenvalue. Denote $\widetilde{\mathbf{x}} = \mathbf{U}^T\mathbf{x}$, $\widetilde{\mathbf{y}} = \mathbf{U}^T\mathbf{y}$, and $\widetilde{\nabla} f(\mathbf{y}) = \mathbf{U}^T\nabla f(\mathbf{y})$. Let $\widetilde{\mathbf{x}}_j$ and $\widetilde{\nabla}_j f(\mathbf{y})$ be the jth element of $\widetilde{\mathbf{x}}$ and $\widetilde{\nabla} f(\mathbf{y})$, respectively. From the ρ-Hessian Lipschitz assumption, we have

$$\begin{aligned}
&f(\mathbf{x}^{\mathcal{K}}) - f(\mathbf{x}^0)\\
\leq&\left\langle \nabla f(\mathbf{x}^0), \mathbf{x}^{\mathcal{K}} - \mathbf{x}^0\right\rangle + \frac{1}{2}(\mathbf{x}^{\mathcal{K}} - \mathbf{x}^0)^T\mathbf{H}(\mathbf{x}^{\mathcal{K}} - \mathbf{x}^0)\\
&+\frac{\rho}{6}\|\mathbf{x}^{\mathcal{K}} - \mathbf{x}^0\|^3\\
=&\left\langle \widetilde{\nabla} f(\mathbf{x}^0), \widetilde{\mathbf{x}}^{\mathcal{K}} - \widetilde{\mathbf{x}}^0\right\rangle + \frac{1}{2}(\widetilde{\mathbf{x}}^{\mathcal{K}} - \widetilde{\mathbf{x}}^0)^T\Lambda(\widetilde{\mathbf{x}}^{\mathcal{K}} - \widetilde{\mathbf{x}}^0)\\
&+\frac{\rho}{6}\|\mathbf{x}^{\mathcal{K}} - \mathbf{x}^0\|^3\\
\leq& g(\widetilde{\mathbf{x}}^{\mathcal{K}}) - g(\widetilde{\mathbf{x}}^0) + 4.5\rho B^3,
\end{aligned} \tag{5}$$

where we denote

$$\begin{aligned}
g(\mathbf{x}) &= \left\langle \widetilde{\nabla} f(\mathbf{x}^0), \mathbf{x} - \widetilde{\mathbf{x}}^0\right\rangle + \frac{1}{2}(\mathbf{x} - \widetilde{\mathbf{x}}^0)^T\Lambda(\mathbf{x} - \widetilde{\mathbf{x}}^0),\\
g_j(x) &= \left\langle \widetilde{\nabla}_j f(\mathbf{x}^0), x - \widetilde{\mathbf{x}}_j^0\right\rangle + \frac{1}{2}\lambda_j(x - \widetilde{\mathbf{x}}_j^0)^2.
\end{aligned}$$

Denoting

$$\widetilde{\delta}_j^k = \widetilde{\nabla}_j f(\mathbf{y}^k) - \nabla g_j(\widetilde{\mathbf{y}}_j^k), \qquad \widetilde{\delta}^k = \widetilde{\nabla} f(\mathbf{y}^k) - \nabla g(\widetilde{\mathbf{y}}^k),$$

then the iterations can be rewritten as

$$\widetilde{\mathbf{y}}_j^k = \widetilde{\mathbf{x}}_j^k + (1-\theta)(\widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}), \tag{6a}$$
$$\widetilde{\mathbf{x}}_j^{k+1} = \widetilde{\mathbf{y}}_j^k - \eta\widetilde{\nabla}_j f(\mathbf{y}^k) = \widetilde{\mathbf{y}}_j^k - \eta\nabla g_j(\widetilde{\mathbf{y}}_j^k) - \eta\widetilde{\delta}_j^k, \tag{6b}$$

and $\|\widetilde{\delta}^k\|$ can be bounded as

$$\begin{aligned}
&\|\widetilde{\delta}^k\|\\
=&\|\widetilde{\nabla} f(\mathbf{y}^k) - \widetilde{\nabla} f(\mathbf{x}^0) - \Lambda(\widetilde{\mathbf{y}}^k - \widetilde{\mathbf{x}}^0)\|\\
=&\|\nabla f(\mathbf{y}^k) - \nabla f(\mathbf{x}^0) - \mathbf{H}(\mathbf{y}^k - \mathbf{x}^0)\|\\
=&\left\|\left(\int_0^1 \nabla^2 f(\mathbf{x}^0 + t(\mathbf{y}^k - \mathbf{x}^0)) - \mathbf{H}\right)(\mathbf{y}^k - \mathbf{x}^0)dt\right\|\\
\leq&\frac{\rho}{2}\|\mathbf{y}^k - \mathbf{x}^0\|^2 \leq 2\rho B^2,
\end{aligned} \tag{7}$$

for any $k < \mathcal{K}$, where we use the ρ-Hessian Lipschitz assumption and (2c) in the last two inequalities.

From (5), to prove the decrease from $f(\mathbf{x}^0)$ to $f(\mathbf{x}^{\mathcal{K}})$, we only need to study $g(\widetilde{\mathbf{x}}^{\mathcal{K}}) - g(\widetilde{\mathbf{x}}^0)$, that is, the decrease of $g(\mathbf{x})$. Iterations (6a) and (6b) can be viewed as applying AGD to the quadratic approximation $g(\mathbf{x})$ coordinately with the approximation error $\widetilde{\delta}^k$, which can be controlled within $\mathcal{O}(\rho B^2)$. The quadratic function $g(\mathbf{x})$ equals to the sum of d scalar functions $g_j(\mathbf{x}_j)$. We decompose $g(\mathbf{x})$ into $\sum_{j\in\mathcal{S}_1} g_j(\mathbf{x}_j)$ and $\sum_{j\in\mathcal{S}_2} g_j(\mathbf{x}_j)$, where

$$\mathcal{S}_1 = \left\{j : \lambda_j \geq -\frac{\theta}{\eta}\right\}, \quad \mathcal{S}_2 = \left\{j : \lambda_j < -\frac{\theta}{\eta}\right\}.$$

We see that $g_j(x)$ is approximate convex when $j \in \mathcal{S}_1$, and strongly concave when $j \in \mathcal{S}_2$.

It is pointed out in (Jin et al., 2018) that the major challenge in analyzing nonconvex momentum-based methods is that the objective function does not decrease monotonically. To overcome this issue, Jin et al. (2018) design a potential function and use the negative curvature exploitation when the objective is very nonconvex to guarantee the decrease of the potential function. An open problem is asked in Section 5 of (Jin et al., 2018) whether the negative curvature exploitation is necessary for the fast rate.

In contrast with (Jin et al., 2018), we establish the approximate decrease of some specified potential function when $j \in \mathcal{S}_1$, as shown in (9), and the approximate decrease of $g_j(x)$ when $j \in \mathcal{S}_2$, given in (12). Thus, the negative curvature exploitation is avoided. Putting the two cases together, we can show the decrease of $f(\mathbf{x})$ in each epoch.

We first consider $\sum_{j\in\mathcal{S}_1} g_j(\mathbf{x}_j)$ in the following lemma.

Lemma 3.2. *Suppose that Assumption 2.1 holds. Let $\eta \leq \frac{1}{4L}$ and $0 \leq \theta \leq 1$. When the "if condition" triggers and $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| \leq \frac{B}{\eta}$, then for Algorithm 1 we have*

$$\begin{aligned}
&\sum_{j\in\mathcal{S}_1} g_j(\widetilde{\mathbf{x}}_j^{\mathcal{K}}) - \sum_{j\in\mathcal{S}_1} g_j(\widetilde{\mathbf{x}}_j^0)\\
\leq& -\sum_{j\in\mathcal{S}_1}\frac{3\theta}{8\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k|^2 + \frac{8\eta\rho^2B^4\mathcal{K}}{\theta}.
\end{aligned} \tag{8}$$

Proof. Since $g_j(x)$ is quadratic, we have

$$
\begin{aligned}
&g_j(\widetilde{\mathbf{x}}_j^{k+1})\\
=&g_j(\widetilde{\mathbf{x}}_j^k)+\left\langle\nabla g_j(\widetilde{\mathbf{x}}_j^k),\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle+\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\\
\overset{a}{=}&g_j(\widetilde{\mathbf{x}}_j^k)-\frac{1}{\eta}\left\langle\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k+\eta\widetilde{\delta}_j^k,\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle\\
&+\left\langle\nabla g_j(\widetilde{\mathbf{x}}_j^k)-\nabla g_j(\widetilde{\mathbf{y}}_j^k),\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle+\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\\
=&g_j(\widetilde{\mathbf{x}}_j^k)-\frac{1}{\eta}\left\langle\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k,\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle-\left\langle\widetilde{\delta}_j^k,\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle\\
&+\lambda_j\left\langle\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k,\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle+\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\\
=&g_j(\widetilde{\mathbf{x}}_j^k)+\frac{1}{2\eta}\left(|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\right)\\
&-\left\langle\widetilde{\delta}_j^k,\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\right\rangle+\frac{\lambda_j}{2}\left(|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2\right)\\
\le&g_j(\widetilde{\mathbf{x}}_j^k)+\frac{1}{2\eta}\left(|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\right)\\
&+\frac{1}{2\alpha}|\widetilde{\delta}_j^k|^2+\frac{\alpha}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{\lambda_j}{2}\left(|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2\right),
\end{aligned}
$$

where we use (6b) in $\overset{a}{=}$. Using $L\ge\lambda_j\ge-\frac{\theta}{\eta}$ when $j\in\mathcal{S}_1=\{j:\lambda_j\ge-\frac{\theta}{\eta}\}$ and $\left(-\frac{1}{2\eta}+\frac{\lambda_j}{2}\right)|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2\le\left(-2L+\frac{L}{2}\right)|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{y}}_j^k|^2\le0$, we have for each $j\in\mathcal{S}_1$,

$$
\begin{aligned}
g_j(\widetilde{\mathbf{x}}_j^{k+1})\le&g_j(\widetilde{\mathbf{x}}_j^k)+\frac{1}{2\eta}\left(|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2-|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\right)\\
&+\frac{1}{2\alpha}|\widetilde{\delta}_j^k|^2+\frac{\alpha}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{\theta}{2\eta}|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{y}}_j^k|^2\\
\overset{b}{=}&g_j(\widetilde{\mathbf{x}}_j^k)+\frac{(1-\theta)^2(1+\theta)}{2\eta}|\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{x}}_j^{k-1}|^2\\
&-\left(\frac{1}{2\eta}-\frac{\alpha}{2}\right)|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{1}{2\alpha}|\widetilde{\delta}_j^k|^2,
\end{aligned}
$$

where we use (6a) in $\overset{b}{=}$. Defining the potential function

$$
\ell_j^{k+1}=g_j(\widetilde{\mathbf{x}}_j^{k+1})+\frac{(1-\theta)^2(1+\theta)}{2\eta}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2,
$$

we have

$$
\begin{aligned}
\ell_j^{k+1}\le&\ell_j^k+\frac{1}{2\alpha}|\widetilde{\delta}_j^k|^2\\
&-\left(\frac{1}{2\eta}-\frac{\alpha}{2}-\frac{(1-\theta)^2(1+\theta)}{2\eta}\right)|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2\\
\overset{c}{\le}&\ell_j^k-\frac{3\theta}{8\eta}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{2\eta}{\theta}|\widetilde{\delta}_j^k|^2,
\end{aligned}
\tag{9}
$$

where we let $\alpha=\frac{\theta}{4\eta}$ in $\overset{c}{\le}$ such that $\frac{1}{2\eta}-\frac{\theta}{8\eta}-\frac{(1-\theta)^2(1+\theta)}{2\eta}=\frac{3\theta}{8\eta}+\frac{\theta^2}{2\eta}-\frac{\theta^3}{2\eta}\ge\frac{3\theta}{8\eta}$. Summing over $k=0,1,\cdots,\mathcal{K}-1$ and $j\in\mathcal{S}_1$, using $\mathbf{x}^0-\mathbf{x}^{-1}=0$, we have

$$
\begin{aligned}
&\sum_{j\in\mathcal{S}_1}g_j(\widetilde{\mathbf{x}}_j^{\mathcal{K}})\le\sum_{j\in\mathcal{S}_1}\ell_j^{\mathcal{K}}\\
\le&\sum_{j\in\mathcal{S}_1}g_j(\widetilde{\mathbf{x}}_j^0)-\sum_{j\in\mathcal{S}_1}\frac{3\theta}{8\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{2\eta}{\theta}\sum_{k=0}^{\mathcal{K}-1}\|\widetilde{\delta}^k\|^2\\
\overset{d}{\le}&\sum_{j\in\mathcal{S}_1}g_j(\widetilde{\mathbf{x}}_j^0)-\sum_{j\in\mathcal{S}_1}\frac{3\theta}{8\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{8\eta\rho^2B^4\mathcal{K}}{\theta},
\end{aligned}
$$

where we use (7) in $\overset{d}{\le}$. □

Next, we consider $\sum_{j\in\mathcal{S}_2}g_j(\mathbf{x}_j)$.

Lemma 3.3. *Suppose that Assumption 2.1 holds. Let $\eta\le\frac{1}{4L}$ and $0\le\theta\le1$. When the "if condition" triggers and $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\|\le\frac{B}{\eta}$, then for Algorithm 1 we have*

$$
\begin{aligned}
&\sum_{j\in\mathcal{S}_2}g_j(\widetilde{\mathbf{x}}_j^{\mathcal{K}})-\sum_{j\in\mathcal{S}_2}g_j(\widetilde{\mathbf{x}}_j^0)\\
\le&-\sum_{j\in\mathcal{S}_2}\frac{\theta}{2\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\frac{2\eta\rho^2B^4\mathcal{K}}{\theta}.
\end{aligned}
\tag{10}
$$

Proof. Denoting $\mathbf{v}_j=\widetilde{\mathbf{x}}_j^0-\frac{1}{\lambda_j}\widetilde{\nabla}_jf(\mathbf{x}^0)$, $g_j(x)$ can be rewritten as

$$
\begin{aligned}
g_j(x)=&\frac{\lambda_j}{2}\left(x-\widetilde{\mathbf{x}}_j^0+\frac{1}{\lambda_j}\widetilde{\nabla}_jf(\mathbf{x}^0)\right)^2-\frac{1}{2\lambda_j}|\widetilde{\nabla}_jf(\mathbf{x}^0)|^2\\
=&\frac{\lambda_j}{2}|x-\mathbf{v}_j|^2-\frac{1}{2\lambda_j}|\widetilde{\nabla}_jf(\mathbf{x}^0)|^2.
\end{aligned}
$$

For each $j\in\mathcal{S}_2=\{j:\lambda_j<-\frac{\theta}{\eta}\}$, we have

$$
\begin{aligned}
&g_j(\widetilde{\mathbf{x}}_j^{k+1})-g_j(\widetilde{\mathbf{x}}_j^k)\\
=&\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\mathbf{v}_j|^2-\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^k-\mathbf{v}_j|^2\\
=&\frac{\lambda_j}{2}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\lambda_j\left\langle\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k,\widetilde{\mathbf{x}}_j^k-\mathbf{v}_j\right\rangle\\
\le&-\frac{\theta}{2\eta}|\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k|^2+\lambda_j\left\langle\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k,\widetilde{\mathbf{x}}_j^k-\mathbf{v}_j\right\rangle.
\end{aligned}
\tag{11}
$$

So we only need to bound the second term. From (6b) and (6a), we have

$$
\begin{aligned}
&\widetilde{\mathbf{x}}_j^{k+1}-\widetilde{\mathbf{x}}_j^k\\
=&\widetilde{\mathbf{y}}_j^k-\widetilde{\mathbf{x}}_j^k-\eta\nabla g_j(\widetilde{\mathbf{y}}_j^k)-\eta\widetilde{\delta}_j^k\\
=&(1-\theta)(\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{x}}_j^{k-1})-\eta\nabla g_j(\widetilde{\mathbf{y}}_j^k)-\eta\widetilde{\delta}_j^k\\
=&(1-\theta)(\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{x}}_j^{k-1})-\eta\lambda_j(\widetilde{\mathbf{y}}_j^k-\mathbf{v}_j)-\eta\widetilde{\delta}_j^k\\
=&(1-\theta)(\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{x}}_j^{k-1})\\
&-\eta\lambda_j(\widetilde{\mathbf{x}}_j^k-\mathbf{v}_j+(1-\theta)(\widetilde{\mathbf{x}}_j^k-\widetilde{\mathbf{x}}_j^{k-1}))-\eta\widetilde{\delta}_j^k.
\end{aligned}
$$

So for each $j \in \mathcal{S}_2$, we have

$$
\begin{aligned}
&\left\langle \widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle \\
=&(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle - \eta\lambda_j|\widetilde{\mathbf{x}}_j^k - \mathbf{v}_j|^2 \\
&-\eta\lambda_j(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle - \eta\left\langle \widetilde{\delta}_j^k, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle \\
\overset{a}{\geq}&(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle - \eta\lambda_j|\widetilde{\mathbf{x}}_j^k - \mathbf{v}_j|^2 \\
&+\frac{\eta\lambda_j(1-\theta)}{2}\left(|\widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}|^2 + |\widetilde{\mathbf{x}}_j^k - \mathbf{v}_j|^2\right) \\
&+\frac{\eta}{2\lambda_j(1+\theta)}|\widetilde{\delta}_j^k|^2 + \frac{\eta\lambda_j(1+\theta)}{2}|\widetilde{\mathbf{x}}_j^k - \mathbf{v}_j|^2 \\
=&(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle \\
&+\frac{\eta\lambda_j(1-\theta)}{2}|\widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}|^2 + \frac{\eta}{2\lambda_j(1+\theta)}|\widetilde{\delta}_j^k|^2 \\
=&(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^{k-1} - \mathbf{v}_j \right\rangle + (1-\theta)|\widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}|^2 \\
&+\frac{\eta\lambda_j(1-\theta)}{2}|\widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}|^2 + \frac{\eta}{2\lambda_j(1+\theta)}|\widetilde{\delta}_j^k|^2 \\
\overset{b}{\geq}&(1-\theta)\left\langle \widetilde{\mathbf{x}}_j^k - \widetilde{\mathbf{x}}_j^{k-1}, \widetilde{\mathbf{x}}_j^{k-1} - \mathbf{v}_j \right\rangle + \frac{\eta}{2\lambda_j}|\widetilde{\delta}_j^k|^2,
\end{aligned}
$$

where we use the fact that $\lambda_j < 0$ when $j \in \mathcal{S}_2$ in $\overset{a}{\geq}$ and $\left(1+\frac{\eta\lambda_j}{2}\right)(1-\theta) \geq \left(1-\frac{\eta L}{2}\right)(1-\theta) \geq 0$ in $\overset{b}{\geq}$. So we have

$$
\begin{aligned}
&\left\langle \widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k, \widetilde{\mathbf{x}}_j^k - \mathbf{v}_j \right\rangle \\
\geq&(1-\theta)^k\left\langle \widetilde{\mathbf{x}}_j^1 - \widetilde{\mathbf{x}}_j^0, \widetilde{\mathbf{x}}_j^0 - \mathbf{v}_j \right\rangle + \frac{\eta}{2\lambda_j}\sum_{t=1}^k(1-\theta)^{k-t}|\widetilde{\delta}_j^t|^2 \\
\overset{c}{=}&-(1-\theta)^k\eta\lambda_j|\widetilde{\mathbf{x}}_j^0 - \mathbf{v}_j|^2 + \frac{\eta}{2\lambda_j}\sum_{t=1}^k(1-\theta)^{k-t}|\widetilde{\delta}_j^t|^2 \\
\overset{d}{\geq}&\frac{\eta}{2\lambda_j}\sum_{t=1}^k(1-\theta)^{k-t}|\widetilde{\delta}_j^t|^2,
\end{aligned}
$$

where we use

$$
\begin{aligned}
\widetilde{\mathbf{x}}_j^1 - \widetilde{\mathbf{x}}_j^0 =& \widetilde{\mathbf{x}}_j^1 - \widetilde{\mathbf{y}}_j^0 = -\eta\widetilde{\nabla}_j f(\mathbf{y}^0) = -\eta\widetilde{\nabla}_j f(\mathbf{x}^0) \\
=& -\eta\nabla g_j(\widetilde{\mathbf{x}}_j^0) = -\eta\lambda_j(\widetilde{\mathbf{x}}_j^0 - \mathbf{v}_j)
\end{aligned}
$$

in $\overset{c}{=}$ and $\lambda_j < 0$ in $\overset{d}{\geq}$. Plugging into (11) and using $\lambda_j < 0$ again, we have

$$
\begin{aligned}
&g_j(\widetilde{\mathbf{x}}_j^{k+1}) - g_j(\widetilde{\mathbf{x}}_j^k) \\
\leq& -\frac{\theta}{2\eta}|\widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k|^2 + \frac{\eta}{2}\sum_{t=1}^k(1-\theta)^{k-t}|\widetilde{\delta}_j^t|^2.
\end{aligned} \tag{12}
$$

Summing over $k = 0, 1, \cdots, \mathcal{K}-1$ and $j \in \mathcal{S}_2$, we have

$$
\begin{aligned}
&\sum_{j\in\mathcal{S}_2} g_j(\widetilde{\mathbf{x}}_j^{\mathcal{K}}) - \sum_{j\in\mathcal{S}_2} g_j(\widetilde{\mathbf{x}}_j^0) \\
\leq& -\sum_{j\in\mathcal{S}_2}\frac{\theta}{2\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k|^2 + \frac{\eta}{2}\sum_{k=0}^{\mathcal{K}-1}\sum_{t=1}^k(1-\theta)^{k-t}\|\widetilde{\delta}^t\|^2 \\
\overset{e}{\leq}& -\sum_{j\in\mathcal{S}_2}\frac{\theta}{2\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k|^2 + 2\eta\rho^2B^4\sum_{k=0}^{\mathcal{K}-1}\sum_{t=1}^k(1-\theta)^{k-t} \\
\leq& -\sum_{j\in\mathcal{S}_2}\frac{\theta}{2\eta}\sum_{k=0}^{\mathcal{K}-1}|\widetilde{\mathbf{x}}_j^{k+1} - \widetilde{\mathbf{x}}_j^k|^2 + \frac{2\eta\rho^2B^4\mathcal{K}}{\theta},
\end{aligned}
$$

where we use (7) in $\overset{e}{\leq}$. □

Putting Lemmas 3.2 and 3.3 together, we can show the decrease of $f(\mathbf{x})$ in each epoch.

Lemma 3.4. *Suppose that Assumption 2.1 holds. Under the parameter settings in Theorem 2.2, when the "if condition" triggers and $\|\nabla f(\mathbf{y}^{\mathcal{K}-1})\| \leq \frac{B}{\eta}$, then for Algorithm 1 we have*

$$
f(\mathbf{x}^{\mathcal{K}}) - f(\mathbf{x}^0) \leq -\frac{\epsilon^{3/2}}{\sqrt{\rho}}.
$$

Proof. Summing over (8) and (10), we have

$$
\begin{aligned}
g(\widetilde{\mathbf{x}}^{\mathcal{K}}) - g(\widetilde{\mathbf{x}}^0) =& \sum_{j\in\mathcal{S}_1\cup\mathcal{S}_2} g_j(\widetilde{\mathbf{x}}_j^{\mathcal{K}}) - g_j(\widetilde{\mathbf{x}}_j^0) \\
\leq& -\frac{3\theta}{8\eta}\sum_{k=0}^{\mathcal{K}-1}\|\widetilde{\mathbf{x}}^{k+1} - \widetilde{\mathbf{x}}^k\|^2 + \frac{10\eta\rho^2B^4\mathcal{K}}{\theta} \\
=& -\frac{3\theta}{8\eta}\sum_{k=0}^{\mathcal{K}-1}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \frac{10\eta\rho^2B^4\mathcal{K}}{\theta} \\
\overset{a}{\leq}& -\frac{3\theta B^2}{8\eta\mathcal{K}} + \frac{10\eta\rho^2B^4\mathcal{K}}{\theta},
\end{aligned}
$$

where we use (2a) in $\overset{a}{\leq}$. Plugging into (5) and using $\mathcal{K} \leq K$, we have

$$
\begin{aligned}
&f(\mathbf{x}^{\mathcal{K}}) - f(\mathbf{x}^0) \\
\leq& -\frac{3\theta B^2}{8\eta\mathcal{K}} + \frac{10\rho^2B^4\eta\mathcal{K}}{2\theta} + 4.5\rho B^3 \\
\leq& -\frac{3\theta B^2}{8\eta K} + \frac{10\rho^2B^4\eta K}{2\theta} + 4.5\rho B^3 \leq -\frac{\epsilon^{3/2}}{\sqrt{\rho}}.
\end{aligned} \tag{13}
$$

□

3.3. Small Gradient in the Last Epoch

In this section, we prove Theorem 2.2. The main job is to establish $\|\nabla f(\hat{\mathbf{y}})\| \leq \mathcal{O}(\epsilon)$ in the last epoch.

Proof. From Lemmas 3.1 and 3.4, we have

$$f(\mathbf{x}^{\mathcal{K}})-f(\mathbf{x}^0)\leq-\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}. \tag{14}$$

Note that at the beginning of each epoch in Algorithm 1, we set $\mathbf{x}^0$ to be the last iterate $\mathbf{x}^{\mathcal{K}}$ in the previous epoch. Summing (14) over all epochs, say N total epochs, we have

$$\min_{\mathbf{x}} f(\mathbf{x})-f(\mathbf{x}_{int})\leq -N\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}.$$

So the algorithm will terminate in at most $\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}}$ epochs. Since each epoch needs at most $K=\frac{1}{2}\left(\frac{L^2}{\epsilon\rho}\right)^{1/4}$ gradient evaluations, the total number of gradient evaluations must be less than $\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}$.

Now, we consider the last epoch. Denote $\widetilde{\mathbf{y}}=\mathbf{U}^T\hat{\mathbf{y}}=\frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{U}^T\mathbf{y}^k = \frac{1}{K_0+1}\sum_{k=0}^{K_0}\widetilde{\mathbf{y}}^k$. Since g is quadratic, we have

$$\begin{aligned}
&\|\nabla g(\widetilde{\mathbf{y}})\|=\left\|\frac{1}{K_0+1}\sum_{k=0}^{K_0}\nabla g(\widetilde{\mathbf{y}}^k)\right\|\\
\overset{a}{=}&\frac{1}{\eta(K_0+1)}\left\|\sum_{k=0}^{K_0}\left(\widetilde{\mathbf{x}}^{k+1}-\widetilde{\mathbf{y}}^k+\eta\widetilde{\delta}^k\right)\right\|\\
=&\frac{1}{\eta(K_0+1)}\left\|\sum_{k=0}^{K_0}\left(\widetilde{\mathbf{x}}^{k+1}-\widetilde{\mathbf{x}}^k-(1-\theta)(\widetilde{\mathbf{x}}^k-\widetilde{\mathbf{x}}^{k-1})+\eta\widetilde{\delta}^k\right)\right\|\\
\overset{b}{=}&\frac{1}{\eta(K_0+1)}\left\|\widetilde{\mathbf{x}}^{K_0+1}-\widetilde{\mathbf{x}}^0-(1-\theta)(\widetilde{\mathbf{x}}^{K_0}-\widetilde{\mathbf{x}}^0)+\eta\sum_{k=0}^{K_0}\widetilde{\delta}^k\right\|\\
=&\frac{1}{\eta(K_0+1)}\left\|\widetilde{\mathbf{x}}^{K_0+1}-\widetilde{\mathbf{x}}^{K_0}+\theta(\widetilde{\mathbf{x}}^{K_0}-\widetilde{\mathbf{x}}^0)+\eta\sum_{k=0}^{K_0}\widetilde{\delta}^k\right\|\\
\leq&\frac{1}{\eta(K_0+1)}\left(\|\widetilde{\mathbf{x}}^{K_0+1}-\widetilde{\mathbf{x}}^{K_0}\|+\theta\|\widetilde{\mathbf{x}}^{K_0}-\widetilde{\mathbf{x}}^0\|+\eta\sum_{k=0}^{K_0}\|\widetilde{\delta}^k\|\right)\\
\overset{c}{\leq}&\frac{2}{\eta K}\|\widetilde{\mathbf{x}}^{K_0+1}-\widetilde{\mathbf{x}}^{K_0}\|+\frac{2\theta B}{\eta K}+2\rho B^2,
\end{aligned} \tag{15}$$

where we use (6b) in $\overset{a}{=}$, $\mathbf{x}^{-1}=\mathbf{x}^0$ in $\overset{b}{=}$, $K_0+1\geq\frac{K}{2}$, (3a), (7), and (3b) in $\overset{c}{\leq}$. From $K_0=\operatorname{argmin}_{\lfloor\frac{K}{2}\rfloor\leq k\leq K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$, we have

$$\begin{aligned}
&\|\mathbf{x}^{K_0+1}-\mathbf{x}^{K_0}\|^2\\
\leq&\frac{1}{K-\lfloor K/2\rfloor}\sum_{k=\lfloor K/2\rfloor}^{K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\\
\leq&\frac{1}{K-\lfloor K/2\rfloor}\sum_{k=0}^{K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\overset{d}{\leq}\frac{2B^2}{K^2},
\end{aligned} \tag{16}$$

where we use (3a) in $\overset{d}{\leq}$. On the other hand, we also have

$$\begin{aligned}
\|\nabla f(\hat{\mathbf{y}})\|=&\|\widetilde{\nabla} f(\hat{\mathbf{y}})\|\leq\|\nabla g(\widetilde{\mathbf{y}})\|+\|\widetilde{\nabla} f(\hat{\mathbf{y}})-\nabla g(\widetilde{\mathbf{y}})\|\\
=&\|\nabla g(\widetilde{\mathbf{y}})\|+\|\widetilde{\nabla} f(\hat{\mathbf{y}})-\widetilde{\nabla} f(\mathbf{x}^0)-\Lambda(\widetilde{\mathbf{y}}-\widetilde{\mathbf{x}}^0)\|\\
=&\|\nabla g(\widetilde{\mathbf{y}})\|+\|\nabla f(\hat{\mathbf{y}})-\nabla f(\mathbf{x}^0)-\mathbf{H}(\hat{\mathbf{y}}-\mathbf{x}^0)\|\\
\leq&\|\nabla g(\widetilde{\mathbf{y}})\|+\frac{\rho}{2}\|\hat{\mathbf{y}}-\mathbf{x}^0\|^2\overset{e}{\leq}\|\nabla g(\widetilde{\mathbf{y}})\|+2\rho B^2,
\end{aligned}$$

where we use $\|\hat{\mathbf{y}}-\mathbf{x}^0\|\leq\frac{1}{K_0+1}\sum_{k=0}^{K_0}\|\mathbf{y}^k-\mathbf{x}^0\|\leq 2B$ from (3b) in $\overset{e}{\leq}$. So we have

$$\|\nabla f(\hat{\mathbf{y}})\|\leq\frac{2\sqrt{2}B}{\eta K^2}+\frac{2\theta B}{\eta K}+4\rho B^2\leq 82\epsilon.$$

□

Remark 3.5. The purpose of using $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2>B^2$ in the "if condition", rather than $\|\mathbf{x}^k-\mathbf{x}^0\|\geq B$, and the special average as the output in Algorithm 1 is to establish (16).

3.4. Discussion on the Acceleration Mechanism

When we replace the AGD iterations in Algorithm 1 by the gradient descent iterations $\mathbf{x}^{k+1}=\mathbf{x}^k-\eta\nabla f(\mathbf{x}^k)$ with $\eta=\frac{1}{4L}$, similar to (4), the descent property in each epoch becomes

$$f(\mathbf{x}^{\mathcal{K}})-f(\mathbf{x}^0)\leq-\frac{7}{8\eta}\sum_{k=0}^{\mathcal{K}-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\leq-\frac{7B^2}{8\eta\mathcal{K}},$$

and the gradient norm at the averaged output $\hat{\mathbf{x}}=\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{x}^k$ is bounded as

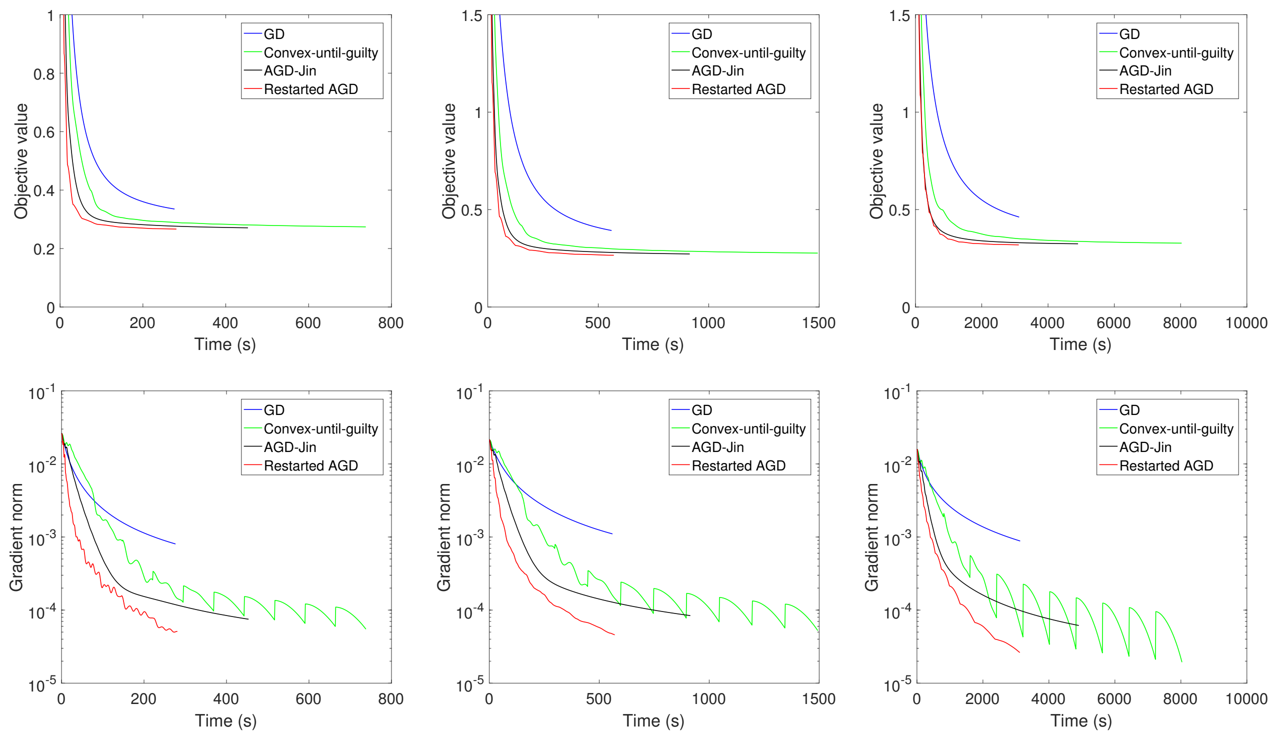
$$\|\nabla g(\hat{\mathbf{x}})\|\leq\frac{1}{\eta K}\|\mathbf{x}^K-\mathbf{x}^0\|+2\rho B^2\leq\frac{B}{\eta K}+2\rho B^2.$$

By setting $B=\sqrt{\frac{\epsilon}{\rho}}$ and $K=\frac{L}{\sqrt{\epsilon\rho}}$, we have the $\mathcal{O}(\epsilon^{-2})$ complexity.

Comparing with (13) and (15), respectively, we see that the momentum parameter θ is crucial to speedup the convergence because it allows smaller K, that is, $\frac{1}{\epsilon^{1/4}}$ v.s. $\frac{1}{\epsilon^{1/2}}$ for AGD and GD, respectively. Accordingly, smaller K results in less total gradient computations. Thus, the acceleration mechanism for nonconvex optimization seems irrelevant to the analysis of convex AGD. It is just because of the momentum.

4. Experiments

We test on the matrix completion problem (Negahban & Wainwright, 2012) to verify the efficiency of the proposed method. The goal of matrix completion is to recover the low



(a) MovieLens-10M (b) MovieLens-20M (c) Netflix

Figure 1. Comparisons of function value and gradient norm on the matrix completion problem.

rank matrix from a set of randomly observed entries, which can be formulated as follows:

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}} \frac{1}{2N}\sum_{(i,j)\in\mathcal{O}}(\mathbf{X}_{i,j}-\mathbf{X}^*_{i,j})^2, \quad s.t. \quad \mathrm{rank}(\mathbf{X})\leq r,$$

where $\mathcal{O}$ is the set of observed entries, N is the size of $\mathcal{O}$, and $\mathbf{X}^*$ is the true low rank matrix. We reformulate the above problem in the following matrix factorization form:

$$\min_{\mathbf{U}\in\mathbb{R}^{m\times r},\mathbf{V}\in\mathbb{R}^{n\times r}} \frac{1}{2N}\sum_{(i,j)\in\mathcal{O}}((\mathbf{U}\mathbf{V}^T)_{i,j}-\mathbf{X}^*_{i,j})^2 + \frac{1}{2N}\|\mathbf{U}^T\mathbf{U}-\mathbf{V}^T\mathbf{V}\|_F^2,$$

where r is the rank of $\mathbf{X}^*$ and the regularization is used to balance $\mathbf{U}$ and $\mathbf{V}$.

We test the performance on the Movielen-10M, Movielen-20M and Netflix data sets, where the corresponding observed matrices are of size 69878×10677, 138493×26744, and 480189×17770, respectively. We set $r=10$ and compare restarted AGD (Algorithm 2 in Appendix A) with Jin's AGD (Jin et al., 2018), the 'convex until guilty' method (Carmon et al., 2017), and gradient descent (GD). Denote $\mathbf{X}_{\mathcal{O}}$ to be the observed data and $\mathbf{A}\Sigma\mathbf{B}^T$ to be its SVD. We initialize $\mathbf{U}=\mathbf{A}_{:,1:r}\sqrt{\Sigma_{1:r,1:r}}$ and $\mathbf{V}=\mathbf{B}_{:,1:r}\sqrt{\Sigma_{1:r,1:r}}$ for all the compared methods. It is efficient to compute the maximal r singular values and the corresponding singular vectors of sparse matrices, for example, by Lanczos. We tune the best stepsize η for each compared method on each dataset. Since the Hessian Lipschitz constant ρ is unknown, we set it as 1 for simplicity. For restarted AGD, we follow Theorem 2.2 to set $\epsilon=10^{-16}$, $B=\sqrt{\frac{\epsilon}{\rho}}$, $\theta=4(\epsilon\rho\eta^2)^{1/4}$, and $K=1/\theta$. We set $B_0=100$ and $c=2$ in Algorithm 2. For Jin's AGD (see Algorithm 4 in Section C for example), we set $\theta=4(\epsilon\rho\eta^2)^{1/4}$, $\gamma=\frac{\theta^2}{\eta}$, and $s=\frac{\gamma}{4\rho}$. For the 'convex until guilty' method, we follow the theory in (Carmon et al., 2017) to set the parameters except that we terminate the inner loop after 100 iterations to improve its practical performance. We run each method for 1000 total iterations.

Figure 1 plots the results. We measure the objective function value and gradient norm at each iterate $\mathbf{y}^k$ for restarted AGD and Jin's AGD. We observed that the figures are almost the same when measured at $\mathbf{y}^k$ and $\mathbf{x}^k$ when preparing the experiments. We see that the accelerated methods perform better than GD, which verifies the efficiency of acceleration in nonconvex optimization. We also observe that our method decreases the objective value and gradient norm fastest. GD and our restarted AGD only perform one gradient computation at each iteration, while Jin's AGD and the 'convex until guilty' method need to evaluate additional objective function values. Thus, Their methods need more total running time when we run all the methods for 1000 iterations.

5. Conclusion

This paper proposes a simple restarted AGD for general nonconvex problems under the gradient Lipschitz and Hessian Lipschitz assumptions. Our simple method finds an ϵ-approximate first-order stationary point in $\mathcal{O}(\epsilon^{-7/4})$ gradient computations with simple proofs, which improves over the state-of-the-art complexity by the $\mathcal{O}(\log \frac{1}{\epsilon})$ factor. We hope our analysis may lead to a better understanding of the acceleration mechanism for nonconvex optimization.

6. Acknowledgement

Huan Li is sponsored by NSF China (grant no. 62006116), Jiangsu Province Basic Research Program (grant no. BK20200439), and Project 2019KB0AB02 supported by Zhejiang laboratory. Zhouchen Lin is supported by the NSF China (No. 61731018), NSFC Tianyuan Fund for Mathematics (No. 12026606), and Project 2020BD006 supported by PKU-Baidu Fund.

References

Agarwal, N., Allen-Zhu, Z., Bullins, B., Hazan, E., and Ma, T. Finding approximate local minima for nonconvex optimization in linear time. In *ACM Symposium on the Theory of Computing (STOC)*, pp. 1195–1199, 2017.

Allen-Zhu, Z. and Li, Y. Neon2: Finding local minima via first-order oracles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3716–3726, 2018.

Carmon, Y. and Duchi, J. Analysis of krylov subspace solutions of regularized nonconvex quadratic problems. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 10728–10738, 2018.

Carmon, Y. and Duchi, J. First-order methods for nonconvex quadratic minimization. *SIAM Review*, 62(2):395–436, 2020.

Carmon, Y., Duchi, J. C., Hinder, O., and Sidford, A. Convex until proven guilty: Dimension-free acceleration of gradient descent on non-convex functions. In *International Conference on Machine Learning (ICML)*, pp. 654–663, 2017.

Carmon, Y., Duchi, J. C., Hinder, O., and Sidford, A. Accelerated methods for nonconvex optimization. *SIAM Journal on Optimization*, 28(2):1751–1772, 2018.

Carmon, Y., Duchi, J., Hinder, O., and Sidford, A. Lower bounds for finding stationary points I. *Mathematical Programming*, 184:71–120, 2020.

Carmon, Y., Duchi, J., Hinder, O., and Sidford, A. Lower bounds for finding stationary points II: First-order methods. *Mathematical Programming*, 185:315–355, 2021.

Cartis, C., Gould, N. I. M., and Toint, P. L. On the complexity of sttpest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems. *SIAM Journal on Optimization*, 20(6):2833–2852, 2010.

Davenport, M., Plan, Y., Berg, E. V. D., and Wootters, M. 1-bit matrix completion. *Information and Inference*, 3(3): 189–223, 2014.

Fang, C., Lin, Z., and Zhang, T. Sharp analysis for nonconvex SGD escaping from saddle points. In *Conference On Learning Theory (COLT)*, pp. 1192–1234, 2019.

Ghadimi, S. and Lan, G. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156:59–99, 2016.

Hoffman, A. J. and Wielandt, H. W. The variation of the spectrum of a normal matrix. *Duke Mathematical Journal*, 20:37–39, 1953.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. How to escape saddle points efficiently. In *International Conference on Machine Learning (ICML)*, pp. 1724–1732, 2017.

Jin, C., Netrapalli, P., and Jordan, M. I. Accelerated gradient descent escapes saddle points faster than gradient descent. In *Conference On Learning Theory (COLT)*, pp. 1042–1085, 2018.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature*, 521(7553):436–444, 2015.

Lee, J. D., Simchowitz, M., Jordan, M. I., and Recht, B. Gradient descent only converges to minimizers. In *Conference On Learning Theory (COLT)*, pp. 1246–1257, 2016.

Li, H. and Lin, Z. Accelerated proximal gradient methods for nonconvex programming. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 379–387, 2015.

Li, Q., Zhou, Y., Liang, Y., and Varshney, P. K. Convergence analysis of proximal gradient with momentum for nonconvex optimization. In *International Conference on Machine Learning (ICML)*, pp. 2111–2119, 2017.

Negahban, S. and Wainwright, M. Resticted strong convexity and weighted matrix completion: optimal bounds with noise. *Journal of Machine Learning Research*, 13(53): 1665–1697, 2012.

Nesterov, Y. A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Nesterov, Y. On an approach to the construction of optimal methods of minimization of smooth convex functions. *Ekonomika I Mateaticheskie Metody*, 24(3):509–517, 1988.

Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Science+Business Media, 2004.

Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103:127–152, 2005.

O'Neill, M. and Wright, S. J. Behavior of accelerated gradient methods near critical points of nonconvex functions. *Mathematical Programming*, 176:403–427, 2019.

Royer, C. W. and Wright, S. J. Complexity analysis of second-order line-search algorithms for smooth nonconvex optimization. *SIAM Journal on Optimization*, 28(2): 1448–1477, 2018.

Royer, C. W., O'Neill, M., and Wright, S. J. A Newton-CG algorithm with complexity guarantees for smooth unconstrained optimization. *Mathematical Programming*, 180:451–488, 2020.

Xu, Y., Jin, R., and Yang, T. First-order stochastic algorithms for escaping from saddle points in almost linear time. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5535–5545, 2018.

A. Practical Implementation of the Restarted AGD

In Algorithm 1, we set B of the order $\sqrt{\epsilon}$ such that the method may restart frequently in the first few iterations. In this case, Algorithm 1 almost reduces to the classical gradient descent. To make use of the practical superiority of AGD in the first few iterations, we should reduce the frequency of restart. A practical implementation is presented in Algorithm 2, which relaxes the restart condition of $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2$ in Algorithm 1 to $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > \max\{B^2, B_0^2\}$, and B_0 can be initialized much larger than B. We decrease B_0 and drop the whole iterates in this round of AGD when the objective value does not decrease or decreases less than a threshold of the order $\epsilon^{3/2}$. When $B_0 \le B$, Algorithm 2 is equivalent to Algorithm 1. On the other hand, we output the one of $\mathbf{x}^K$ and $\hat{\mathbf{y}}$ with the smaller gradient norm, because in practice we always use the last iterate, rather than the averaged iterate. We present the $\mathcal{O}(\epsilon^{-7/4})$ complexity of Algorithm 2 in Theorem A.1.

Algorithm 2 Practical Restarted AGD

Initialize $\mathbf{x}^{-1}=\mathbf{x}^0=\mathbf{x}^0_{cur}=\mathbf{x}_{int}$, $k=0$, B_0, $c>1$.
while $k<K$ or $B_0>B$ **do**
  $\mathbf{y}^k=\mathbf{x}^k+(1-\theta)(\mathbf{x}^k-\mathbf{x}^{k-1})$
  $\mathbf{x}^{k+1}=\mathbf{y}^k-\eta\nabla f(\mathbf{y}^k)$
  $k=k+1$
  if $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2>\max\{B^2,B_0^2\}$ or $k>K$ **then**
    if $f(\mathbf{x}^k)-f(\mathbf{x}^0)\le-\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}$ **then**
      $\mathbf{x}^{-1}=\mathbf{x}^0=\mathbf{x}^k$, $\mathbf{x}^0_{cur}=\mathbf{x}^k$, $k=0$
    else
      $\mathbf{x}^{-1}=\mathbf{x}^0=\mathbf{x}^0_{cur}$, $k=0$, $B_0=B_0/c$
    end if
  end if
end while
$K_0=\operatorname{argmin}_{\lfloor\frac{K}{2}\rfloor\le k\le K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$
$\hat{\mathbf{y}}=\frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{y}^k$
Output $\mathbf{x}_{out}=\operatorname{argmin}_{\mathbf{x}^K,\hat{\mathbf{y}}}\{\|\nabla f(\mathbf{x}^K)\|,\|\nabla f(\hat{\mathbf{y}})\|\}$

Theorem A.1. *Suppose that Assumption 2.1 holds and use the parameter settings in Theorem 2.2. Then Algorithm 2 terminates in at most $\mathcal{O}\left(\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}\right)$ gradient computations and $\mathcal{O}\left(\frac{\triangle_f\rho^{1/2}}{\epsilon^{3/2}}\right)$ function evaluations, and the output satisfies $\|\nabla f(\mathbf{x}_{out})\|\le 82\epsilon$, where $\triangle_f=f(\mathbf{x}_{int})-\min_{\mathbf{x}}f(\mathbf{x})$.*

Proof. Denote one epoch to be valid when the if condition $f(\mathbf{x}^k)-f(\mathbf{x}^0)\le-\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}$ holds. Otherwise, denote the epoch to be invalid. Since each valid epoch decreases the objective at least $\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}$, we have at most $\max\left\{\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}},\frac{\triangle_f\rho}{\epsilon L}\right\}=\mathcal{O}\left(\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}}\right)$ valid epoches. Each epoch, no matter valid or not, needs at most $K+1$ gradient evaluations. On the other hand, we only need $\log_c\frac{B_0}{B}$ invalid epoches to decrease B_0 smaller than B, and Algorithm 2 is equivalent to Algorithm 1 when $B_0\le B$. From (14), we always have $f(\mathbf{x}^k)-f(\mathbf{x}^0)\le-\min\left\{\frac{\epsilon^{3/2}}{\sqrt{\rho}},\frac{\epsilon L}{\rho}\right\}$ when $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2>B^2$. That is, invalid epoch never appears when $B_0\le B$. So we have at most $\log_c\frac{B_0}{B}$ invalid epoches. So the total number of function evaluations and gradient evaluations is $\mathcal{O}\left(\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}}+\log_c\frac{B_0}{B}\right)$ and $\mathcal{O}\left(\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}}+\log_c\frac{B_0}{B}\right)(K+1)=\mathcal{O}\left(\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}\right)$, respectively.

In the last epoch, we have $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2\le B^2$ for all $k\le K$, and the while loop terminates

Algorithm 3 Perturbed Restarted AGD

Initialize $\mathbf{x}^{-1} = \mathbf{x}^0 = \mathbf{x}_{int} + \xi, \xi \sim$Unif$(\mathbb{B}_0(r))$, $k = 0$.
while $k < K$ **do**
  $\mathbf{y}^k = \mathbf{x}^k + (1-\theta)(\mathbf{x}^k - \mathbf{x}^{k-1})$
  $\mathbf{x}^{k+1} = \mathbf{y}^k - \eta\nabla f(\mathbf{y}^k)$
  $k = k+1$
  if $k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2$ **then**
    $\mathbf{x}^{-1} = \mathbf{x}^0 = \mathbf{x}^k + \xi 1_{\|\nabla f(\mathbf{y}^{k-1})\|\leq\frac{B}{\eta}}$, $k = 0$, $\xi \sim$Unif$(\mathbb{B}_0(r))$
  end if
end while
$K_0 = \text{argmin}_{\lfloor\frac{K}{2}\rfloor\leq k\leq K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$
Output $\hat{\mathbf{y}} = \frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{y}^k$

when k equals K. From the proof of Theorem 2.2, we have $\|\nabla f(\hat{\mathbf{y}})\| \leq 82\epsilon$. Since we output $\mathbf{x}_{out} = \text{argmin}_{\mathbf{x}^K,\hat{\mathbf{y}}}\{\|\nabla f(\mathbf{x}^K)\|, \|\nabla f(\hat{\mathbf{y}})\|\}$, we also have $\|\nabla f(\mathbf{x}_{out})\| \leq 82\epsilon$. □

B. Extension to the Second-order Stationary Point

Algorithm 1 can also find ϵ-approximate second-order stationary point, defined as

$$\|\nabla f(\mathbf{x})\| \leq \epsilon, \quad \lambda_{min}(\nabla^2 f(\mathbf{x})) \geq -\sqrt{\epsilon\rho}.$$

We follow (Jin et al., 2017; 2018) to add the perturbations generated uniformly from the ball $\mathbb{B}_0(r)$ with radius r and center 0. The detailed method is presented in Algorithm 3 and the complexity is given in Theorem B.1. We see that Algorithm 3 needs at most $\mathcal{O}(\epsilon^{-7/4}\log\frac{d}{\zeta\epsilon})$ gradient computations to find an ϵ-approximate second-order stationary point with probability at least $1-\zeta$, where d is the dimension of $\mathbf{x}$ in problem (1). This complexity is the same with the one given in (Jin et al., 2018). Comparing with Theorem 2.2, we see that there is a $\mathcal{O}(\log\frac{d}{\zeta\epsilon})$ term. Currently, it is unclear how to remove it.

Theorem B.1. *Suppose that Assumption 2.1 holds. Let* $\chi = \mathcal{O}(\log\frac{d}{\zeta\epsilon}) \geq 1$, $\eta = \frac{1}{4L}$, $B = \frac{1}{288\chi^2}\sqrt{\frac{\epsilon}{\rho}}$, $\theta = \frac{1}{2}\left(\frac{\epsilon\rho}{L^2}\right)^{1/4} < 1$, $K = \frac{2\chi}{\theta}$, $r = \min\{\frac{B}{2}, \frac{\theta B}{20K}, \sqrt{\frac{\theta B^2}{2K}}\} = \mathcal{O}(\epsilon)$. *Then Algorithm 3 terminates in at most* $\mathcal{O}\left(\frac{\triangle_f L^{1/2}\rho^{1/4}\chi^6}{\epsilon^{7/4}}\right)$ *gradient computations and the output satisfies* $\|\nabla f(\hat{\mathbf{y}})\| \leq \epsilon$, *where* $\triangle_f = f(\mathbf{x}_{int}) - \min_{\mathbf{x}} f(\mathbf{x})$. *It also satisfies* $\lambda_{min}(\nabla^2 f(\hat{\mathbf{y}})) \geq -1.011\sqrt{\epsilon\rho}$ *with probability at least* $1-\zeta$.

Theorem B.1 also applies to the perturbed variant of Algorithm 2. In short, the while loop in Algorithm 2 will not terminate until $B_0 \leq B$, and Algorithm 2 reduces to Algorithm 1 when $B_0 \leq B$.

Now, we prove Theorem 3.3.

Proof. Denote $\mathbf{x}^{t,k}$ to be the iterate and ξ^t to be the perturbation in the tth epoch, $\hat{\mathbf{y}}^t = \frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{y}^{t,k}$. When $\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| > \frac{B}{\eta}$ and the "if condition" triggers, we have from Lemma 3.1 that

$$f(\mathbf{x}^{t,\mathcal{K}}) - f(\mathbf{x}^{t,0}) \leq -\frac{B^2}{4\eta}.$$

Since $\mathbf{x}^{t+1,0} = \mathbf{x}^{t,\mathcal{K}}$, we have

$$f(\mathbf{x}^{t+1,0}) - f(\mathbf{x}^{t,0}) \leq -\frac{B^2}{4\eta}.$$

When $\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| \leq \frac{B}{\eta}$ and the "if condition" triggers, we have from Lemma 3.4 that

$$f(\mathbf{x}^{t,\mathcal{K}}) - f(\mathbf{x}^{t,0}) \leq -\frac{3\theta B^2}{8\eta K} + \frac{10\rho^2B^4\eta K}{2\theta} + 4.5\rho B^3.$$

From the L-gradient Lipschitz, we have

$$\begin{aligned}
&f(\mathbf{x}^{t+1,0}) - f(\mathbf{x}^{t,\mathcal{K}})\\
\leq&\left\langle\nabla f(\mathbf{x}^{t,\mathcal{K}}), \mathbf{x}^{t+1,0}-\mathbf{x}^{t,\mathcal{K}}\right\rangle + \frac{L}{2}\|\mathbf{x}^{t+1,0}-\mathbf{x}^{t,\mathcal{K}}\|^2\\
=&\left\langle\nabla f(\mathbf{x}^{t,\mathcal{K}}), \xi^t\right\rangle + \frac{L}{2}\|\xi^t\|^2 \leq \frac{5Br}{4\eta} + \frac{Lr^2}{2} \leq \frac{\theta B^2}{8\eta K},
\end{aligned}$$

where we use

$$\begin{aligned}
&\|\nabla f(\mathbf{x}^{t,\mathcal{K}})\|\\
\leq&\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| + \|\nabla f(\mathbf{x}^{t,\mathcal{K}}) - \nabla f(\mathbf{y}^{t,\mathcal{K}-1})\|\\
\leq&\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| + L\|\mathbf{x}^{t,\mathcal{K}} - \mathbf{y}^{t,\mathcal{K}-1}\|\\
=&\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| + L\eta\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| \leq \frac{B}{\eta} + LB \leq \frac{5B}{4\eta}
\end{aligned}$$

and $\|\xi^t\| \leq r \leq \min\{\frac{\theta B}{20K}, \sqrt{\frac{\theta B^2}{2K}}\}$. So we have

$$\begin{aligned}
f(\mathbf{x}^{t+1,0}) - f(\mathbf{x}^{t,0}) \leq& -\frac{\theta B^2}{4\eta K} + \frac{10\rho^2B^4\eta K}{2\theta} + 4.5\rho B^3\\
\leq& -\frac{\epsilon^{1.5}}{700000\sqrt{\rho}\chi^5}.
\end{aligned}$$

So the algorithm will terminate in at most $\mathcal{O}(\frac{\triangle_f\sqrt{\rho}\chi^5}{\epsilon^{3/2}})$ epochs. Since each epoch needs at most $K = 4\chi\left(\frac{L^2}{\epsilon\rho}\right)^{1/4}$ gradient evaluations, the total number of gradient evaluations must be less than $\mathcal{O}(\frac{\triangle_f L^{1/2}\rho^{1/4}\chi^6}{\epsilon^{7/4}})$.

Now, we consider the last epoch. Denote it to be the Tth epoch. Similar to the proof of Theorem 2.2, we also have

$$\|\nabla f(\hat{\mathbf{y}}^T)\| \leq \frac{2\sqrt{2}B}{\eta K^2} + \frac{2\theta B}{\eta K} + 4\rho B^2 \leq \frac{\epsilon}{\chi^3} \leq \epsilon.$$

On the other hand, we have

$$
\begin{aligned}
&\|\nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\| \\
\leq&\|\nabla f(\hat{\mathbf{y}}^T)\| + \|\nabla f(\hat{\mathbf{y}}^T) - \nabla f(\mathbf{x}^{T-1,\mathcal{K}})\| \\
&+ \|\nabla f(\mathbf{x}^{T-1,\mathcal{K}}) - \nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\| \\
\leq&\|\nabla f(\hat{\mathbf{y}}^T)\| + L\|\hat{\mathbf{y}}^T - \mathbf{x}^{T,0}\| + L\|\mathbf{x}^{T,0} - \mathbf{x}^{T-1,\mathcal{K}}\| \\
&+ L\|\mathbf{x}^{T-1,\mathcal{K}} - \mathbf{y}^{T-1,\mathcal{K}-1}\| \\
\overset{a}{\leq}&\frac{\epsilon}{\chi^2} + 2LB + Lr + \eta L\|\nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\| \\
\leq&\frac{\epsilon}{\chi^2} + 2.5LB + \frac{1}{4}\|\nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\|,
\end{aligned}
$$

where we use $\|\hat{\mathbf{y}}^T - \mathbf{x}^{T,0}\| \leq \frac{1}{K_0+1}\sum_{k=0}^{K_0}\|\mathbf{y}^{T,k} - \mathbf{x}^{T,0}\| \leq 2B$ in $\overset{a}{\leq}$. So we have $\|\nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\| \leq \frac{4\epsilon}{3\chi^2} + \frac{10LB}{3} \leq \frac{4\epsilon}{3\chi^2} + \frac{5B}{6\eta} \leq \frac{B}{\eta}$ by letting $\epsilon \leq \frac{L^2}{576^2\rho}$. In fact, when $\epsilon > \frac{L^2}{576^2\rho}$, we have $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq -L > -576\sqrt{\rho\epsilon}$ for any $\mathbf{x}$. Thus, in the last epoch, we have $\|\nabla f(\mathbf{y}^{T-1,\mathcal{K}-1})\| \leq \frac{B}{\eta}$. This is the reason why perturbation is not needed when $\|\nabla f(\mathbf{y}^{t,\mathcal{K}-1})\| > \frac{B}{\eta}$.

If $\lambda_{min}(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) \geq -\sqrt{\epsilon\rho}$, from the perturbation theory of eigenvalues (Hoffman & Wielandt, 1953), we have for any j,

$$
\begin{aligned}
&|\lambda_j(\nabla^2 f(\hat{\mathbf{y}}^T)) - \lambda_j(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}}))| \\
\leq&\|\nabla^2 f(\hat{\mathbf{y}}^T) - \nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})\|_2 \\
\leq&\rho\|\hat{\mathbf{y}}^T - \mathbf{x}^{T-1,\mathcal{K}}\| \leq \rho\|\hat{\mathbf{y}}^T - \mathbf{x}^{T,0}\| + \rho r \overset{b}{\leq} 3\rho B,
\end{aligned}
$$

and

$$
\begin{aligned}
&\lambda_j(\nabla^2 f(\hat{\mathbf{y}}^T)) \\
\geq&\lambda_j(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) - |\lambda_j(\nabla^2 f(\hat{\mathbf{y}}^T)) - \lambda_j(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}}))| \\
\geq&-\sqrt{\epsilon\rho} - 3\rho B \geq -1.011\sqrt{\epsilon\rho},
\end{aligned}
$$

where we use $\|\hat{\mathbf{y}}^T - \mathbf{x}^{T,0}\| \leq 2B$ in $\overset{b}{\leq}$.

Now, we consider $\lambda_{min}(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) < -\sqrt{\epsilon\rho}$. Define the stuck region within the perturbation ball $\mathbb{B}_{\mathbf{x}^{T-1,\mathcal{K}}}(r)$ to be the set of points starting from which the "if condition" does not trigger in K iterations, that is,

$$
\mathcal{X}=\begin{cases}
\Big\{\mathbf{x} \in \mathbb{B}_{\mathbf{x}^{T-1,\mathcal{K}}}(r) \Big| \{\mathbf{x}^{T,k}\} \text{ is the RAGD iterate with} \\
\mathbf{x}^{T,0} = \mathbf{x} \text{ and } K\sum_{k=0}^{K-1}\|\mathbf{x}^{T,k+1} - \mathbf{x}^{T,k}\|^2 \leq B^2\Big\}, \\
\qquad\qquad \text{if } \lambda_{min}(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) < -\sqrt{\epsilon\rho}, \\
\emptyset, \qquad\qquad \text{othersize.}
\end{cases}
$$

As pointed out in (Jin et al., 2017; 2018), the shape of the stuck region can be very complicated, but its width along the $\mathbf{e}_1$ direction is thin. Similar to Lemma 8 in (Jin et al., 2018), we know from Lemma B.2 that the probability of the starting point $\mathbf{x}^{T,0} = \mathbf{x}^{T-1,\mathcal{K}} + \xi^t$ located in the stuck region $\mathcal{X}$ is less than

$$
\frac{r_0 V_{d-1}(r)}{V_d(r)} \leq \frac{r_0\sqrt{d}}{r} = \zeta,
$$

where we let $r_0 = \frac{\zeta r}{\sqrt{d}}$.

Denote $\mathcal{H}$ to be the random event of $\mathbf{x}^{T,0} \notin \mathcal{X}$ (the location of $\mathbf{x}^{T,0}$ only depends on $\mathbf{x}^{T-1,\mathcal{K}}$ and the random variable ξ^T). When the random event $\mathcal{H}$ happens, we know that if $\lambda_{min}(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) < -\sqrt{\epsilon\rho}$, the "if condition" must trigger. Thus, with probability at least $1-\zeta$ (the random event $\mathcal{H}$ happens), when the "if condition" does not trigger, we have $\lambda_{min}(\nabla^2 f(\mathbf{x}^{T-1,\mathcal{K}})) \geq -\sqrt{\epsilon\rho}$. Thus, the output $\hat{\mathbf{y}}$ satisfies $\lambda_{min}(\nabla^2 f(\hat{\mathbf{y}}^T)) \geq -1.011\sqrt{\epsilon\rho}$ with probability at least $1-\zeta$.

□

Lemma B.2. *Suppose that $\lambda_{min}(\mathbf{H}) < -\sqrt{\epsilon\rho}$, where $\mathbf{H} = \nabla^2 f(\mathbf{x})$. Let $\mathbf{x}'^0$ and $\mathbf{x}''^0$ be at distance at most r from $\mathbf{x}$. Let $\mathbf{x}'^{-1} = \mathbf{x}'^0$, $\mathbf{x}''^{-1} = \mathbf{x}''^0$, and $\mathbf{x}'^0 - \mathbf{x}''^0 = r_0\mathbf{e}_1$, where $\mathbf{e}_1$ is the minimum eigen-direction of $\mathbf{H}$. Under the parameter settings in Theorem B.1, running AGD starting at $\mathbf{x}'^0$ and $\mathbf{x}''^0$, respectively, we have*

$$
\max\left\{K\sum_{k=0}^{K-1}\|\mathbf{x}'^{k+1} - \mathbf{x}'^k\|^2, K\sum_{k=0}^{K-1}\|\mathbf{x}''^{k+1} - \mathbf{x}''^k\|^2\right\} > B^2.
$$

that is, at least one of the iterates triggers the "if condition".

The proof of this lemma is almost the same as that of Lemma 18 in (Jin et al., 2018). We only list the sketch and the details can be found in (Jin et al., 2018).

Proof. Denote $\mathbf{w}^k = \mathbf{x}'^k - \mathbf{x}''^k$. From the update of AGD, we have

$$
\begin{aligned}
\begin{bmatrix}\mathbf{w}^{k+1}\\ \mathbf{w}^k\end{bmatrix} =& \begin{bmatrix}(2-\theta)(\mathbf{I}-\eta\mathbf{H}) & -(1-\theta)(\mathbf{I}-\eta\mathbf{H})\\ \mathbf{I} & 0\end{bmatrix}\begin{bmatrix}\mathbf{w}^k\\ \mathbf{w}^{k-1}\end{bmatrix} \\
&- \eta\begin{bmatrix}(2-\theta)\triangle^k\mathbf{w}^k - (1-\theta)\triangle^k\mathbf{w}^{k-1}\\ 0\end{bmatrix} \\
=&\mathbf{A}\begin{bmatrix}\mathbf{w}^k\\ \mathbf{w}^{k-1}\end{bmatrix} - \eta\begin{bmatrix}\phi^k\\ 0\end{bmatrix} \\
=&\mathbf{A}^{k+1}\begin{bmatrix}\mathbf{w}^0\\ \mathbf{w}^0\end{bmatrix} - \eta\sum_{r=0}^{k}\mathbf{A}^{k-r}\begin{bmatrix}\phi^r\\ 0\end{bmatrix},
\end{aligned}
$$

and

$$
\mathbf{w}^k = [\mathbf{I}, 0]\mathbf{A}^k\begin{bmatrix}\mathbf{w}^0\\ \mathbf{w}^0\end{bmatrix} - \eta[\mathbf{I}, 0]\sum_{r=0}^{k-1}\mathbf{A}^{k-1-r}\begin{bmatrix}\phi^r\\ 0\end{bmatrix},
$$

where $\triangle^k = \int_0^1\left(\nabla^2 f(t\mathbf{y}'^k + (1-t)\mathbf{y}''^k) - \mathbf{H}\right)dt$ and $\phi^k = (2-\theta)\triangle^k\mathbf{w}^k - (1-\theta)\triangle^k\mathbf{w}^{k-1}$.

Assume that none of the iterates $(\mathbf{x}'^0, \mathbf{x}'^1, \cdots, \mathbf{x}'^K)$ and $(\mathbf{x}''^0, \mathbf{x}''^1, \cdots, \mathbf{x}''^K)$ trigger the "if condition", which yield

$$\begin{aligned}&\|\mathbf{x}'^k - \mathbf{x}'^0\| \leq B, \|\mathbf{y}'^k - \mathbf{x}'^0\| \leq 2B, \forall k \leq K,\\&\|\mathbf{x}''^k - \mathbf{x}''^0\| \leq B, \|\mathbf{y}''^k - \mathbf{x}''^0\| \leq 2B, \forall k \leq K.\end{aligned} \tag{17}$$

We have

$$\begin{aligned}\|\triangle^k\| \leq &\rho \max\{\|\mathbf{y}'^k - \mathbf{x}\|, \|\mathbf{y}''^k - \mathbf{x}\|\}\\ \leq&\rho \max\{\|\mathbf{y}'^k - \mathbf{x}'^0\|, \|\mathbf{y}''^k - \mathbf{x}''^0\|\} + \rho r \leq 3\rho B,\\ \|\phi^k\| \leq &6\rho B(\|\mathbf{w}^k\| + \|\mathbf{w}^{k-1}\|).\end{aligned}$$

We can show the following inequality for all $k \leq K$ by induction:

$$\left\|\eta[\mathbf{I}, 0]\sum_{r=0}^{k-1}\mathbf{A}^{k-1-r}\begin{bmatrix}\phi^r\\0\end{bmatrix}\right\| \leq \frac{1}{2}\left\|[\mathbf{I}, 0]\mathbf{A}^k\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|.$$

It is easy to check the base case holds for $k = 0$. Assume the inequality holds for all steps equal to or less than k. Then we have

$$\begin{aligned}&\|\mathbf{w}^k\| \leq \frac{3}{2}\left\|[\mathbf{I}, 0]\mathbf{A}^k\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|,\\&\|\phi^k\| \leq 18\rho B\left\|[\mathbf{I}, 0]\mathbf{A}^k\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|,\end{aligned}$$

by the monotonicity of $\left\|[\mathbf{I}, 0]\mathbf{A}^k\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|$ in k (Lemma 33 in (Jin et al., 2018)). We also have

$$\begin{aligned}&\left\|\eta[\mathbf{I}, 0]\sum_{r=0}^{k}\mathbf{A}^{k-r}\begin{bmatrix}\phi^r\\0\end{bmatrix}\right\| \leq \eta\sum_{r=0}^{k}\left\|[\mathbf{I}, 0]\mathbf{A}^{k-r}\begin{bmatrix}\mathbf{I}\\0\end{bmatrix}\right\|_2\|\phi^r\|\\&\leq 18\rho B\eta\sum_{r=0}^{k}\left\|[\mathbf{I}, 0]\mathbf{A}^{k-r}\begin{bmatrix}\mathbf{I}\\0\end{bmatrix}\right\|_2\left\|[\mathbf{I}, 0]\mathbf{A}^{r}\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|\\&\overset{a}{=} 18\rho B\eta\sum_{r=0}^{k}|a_{k-r}||a_r - b_r|r_0\\&\overset{b}{\leq} 18\rho B\eta\sum_{r=0}^{k}\left(\frac{2}{\theta} + k + 1\right)|a_{k+1} - b_{k+1}|r_0\\&\leq 18\rho B\eta K\left(\frac{2}{\theta} + K\right)\left\|[\mathbf{I}, 0]\mathbf{A}^{k+1}\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\|,\end{aligned}$$

where we define $[a_k, -b_k] = [1, 0]\mathbf{A}_{min}^k$ and $\mathbf{A}_{min} = \begin{bmatrix}(2-\theta)(1-\eta\lambda_{min}) & -(1-\theta)(1-\eta\lambda_{min})\\1 & 0\end{bmatrix}$, $\overset{a}{=}$ uses the fact that $\mathbf{w}^0 = r_0\mathbf{e}_1$ is along the minimum eigenvector direction of $\mathbf{H}$, $\overset{b}{\leq}$uses Lemma 31 in (Jin et al., 2018). From the parameter settings, we have $18\rho B\eta K\left(\frac{2}{\theta} + K\right) \leq \frac{1}{2}$.

Algorithm 4 AGD-Jin

Initialize $\mathbf{x}^0 = \mathbf{x}_{int}$, $\mathbf{v}^0 = 0$, $k = 0$.
while $k < K$ **do**
  $\mathbf{y}^k = \mathbf{x}^k + (1-\theta)\mathbf{v}^k$
  $\mathbf{x}^{k+1} = \mathbf{y}^k - \eta\nabla f(\mathbf{y}^k)$
  $\mathbf{v}^{k+1} = \mathbf{x}^{k+1} - \mathbf{x}^k$
  if $f(\mathbf{x}^k) < f(\mathbf{y}^k) + \langle\nabla f(\mathbf{y}^k), \mathbf{x}^k - \mathbf{y}^k\rangle - \frac{\gamma}{2}\|\mathbf{x}^k - \mathbf{y}^k\|^2$ **then**
    $\mathbf{x}^{k+1} \leftarrow$Negative Curvature Exploitation$(\mathbf{x}^k, \mathbf{v}^k, s)$
    $\mathbf{x}^0 = \mathbf{x}^{k+1}$, $\mathbf{v}^0 = \mathbf{v}^{k+1} = 0$, $k = 0$
  else if $(k+1)\sum_{t=0}^{k}\|\mathbf{x}^{t+1} - \mathbf{x}^t\|^2 > B^2$ **then**
    $\mathbf{x}^0 = \mathbf{x}^{k+1}$, $\mathbf{v}^0 = \mathbf{v}^{k+1}$, $k = 0$
  else
    $k = k + 1$
  end if
end while
$K_1 = \text{argmin}_{1 \leq k \leq \lceil\frac{K}{3}\rceil}\|\mathbf{x}^k - \mathbf{x}^{k-1}\|$
$K_2 = \text{argmin}_{\lfloor\frac{2K}{3}\rfloor \leq k \leq K-1}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|$
Output $\hat{\mathbf{y}} = \frac{1}{K_2 - K_1 + 1}\sum_{k=K_1}^{K_2}\mathbf{y}^k$

Therefore, the induction is proved, which yields

$$\begin{aligned}\|\mathbf{w}^K\| \geq &\left\|[\mathbf{I}, 0]\mathbf{A}^K\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\| - \left\|\eta[\mathbf{I}, 0]\sum_{r=0}^{K-1}\mathbf{A}^{K-1-r}\begin{bmatrix}\phi^r\\0\end{bmatrix}\right\|\\ \geq&\frac{1}{2}\left\|[\mathbf{I}, 0]\mathbf{A}^K\begin{bmatrix}\mathbf{w}^0\\\mathbf{w}^0\end{bmatrix}\right\| = \frac{r_0}{2}|a_K - b_K|\\ \overset{c}{\geq}&\frac{\theta r_0}{4}\left(1 + \frac{\theta}{2}\right)^K \overset{d}{\geq} 5B,\end{aligned}$$

where $\overset{c}{\geq}$ uses Lemma 33 in (Jin et al., 2018) and $\eta\lambda_{min} \leq -\theta^2$, $\overset{d}{\geq}$ uses $K = \frac{2}{\theta}\log\frac{20B}{\theta r_0}$. However, (17) yields

$$\begin{aligned}\|\mathbf{w}^K\| \leq &\|\mathbf{x}'^K - \mathbf{x}'^0\| + \|\mathbf{x}''^K - \mathbf{x}''^0\|\\&+ \|\mathbf{x} - \mathbf{x}'^0\| + \|\mathbf{x} - \mathbf{x}''^0\| \leq 2B + 2r \leq 4B,\end{aligned}$$

which makes a contradiction. Thus the assumption is wrong and we conclude that at least one of the iterates trigger the "if condition".

□

C. Extension to Jin's Method

In this section, we extend our analysis to the method proposed in (Jin et al., 2018), and detail the method in Algorithm 4. No perturbation is added since we do not consider second-order stationary point for simplicity. Except the perturbation and that we specify the stopping criteria and the output, as well as that we rewrite the algorithm in epochs, Algorithm 4 is equivalent to the one in (Jin et al., 2018). However, we give a slightly faster convergence rate by a $\mathcal{O}(\log\frac{1}{\epsilon})$ factor with much simpler proofs.

Algorithm 5 Negative Curvature Exploitation($\mathbf{x}^k, \mathbf{v}^k, s$)

if $\|\mathbf{v}^k\| \geq s$ **then**
  $\mathbf{x}^{k+1} = \mathbf{x}^k$
else
  $\delta = s\mathbf{v}^k/\|\mathbf{v}^k\|$
  $\mathbf{x}^{k+1} = \text{argmin}_{\mathbf{x}^k+\delta, \mathbf{x}^k-\delta} f(\mathbf{x})$
end if
Return $\mathbf{x}^{k+1}$

Define $\mathcal{K} = k+1$ when k resets to 0. Denote the iterations from $k=0$ to $k=\mathcal{K}$ to be one epoch. For each epoch, we have three cases:

1. The negative curvature exploitation (NCE) is employed at the last iteration.
2. The condition $(k+1)\sum_{t=0}^{k}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2$ triggers at the last iteration. Note that in this case, AGD does not restart because $\mathbf{x}^0 - \mathbf{x}^{-1} = \mathbf{v}^0 \neq 0$.
3. None of the above two cases occurs, and the while loop breaks until $k = K$. This is the last epoch.

Define the potential function $\ell^k = f(\mathbf{x}^k) + \frac{1-\theta}{2\eta}\|\mathbf{v}^k\|^2$. We need the following two lemmas, which can be adapted slightly from Lemmas 4 and 5 in (Jin et al., 2018).

Lemma C.1. *Suppose that Assumption 2.1 holds. Let $\eta \leq \frac{1}{2L}$ and $\theta \in [2\eta\gamma, \frac{1}{2}]$. If NCE is not performed at iteration k, then for Algorithm 4 we have $\ell^{k+1} \leq \ell^k - \frac{\theta}{2\eta}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2$.*

Lemma C.2. *Suppose that Assumption 2.1 holds. Let $\theta \leq \frac{1}{2}$. If NCE is performed at iteration k, then for Algorithm 4 we have $\ell^{k+1} \leq \ell^k - \min\left\{\frac{(1-\theta)s^2}{2\eta}, \frac{(\gamma-2\rho s)s^2}{2}\right\}$.*

Set $\gamma = \frac{\theta^2}{\eta}$, $s = \frac{\gamma}{4\rho}$, and the other parameters the same as those in Theorem 2.2. In Case 1, we know from Lemma C.2 that the potential function decreases with a magnitude at least $\frac{64\epsilon^{1.5}}{\sqrt{\rho}}$ at the last iteration, and it does not increase in the previous iterations from Lemma C.1. So we have

$$\ell^{\mathcal{K}} \leq \ell^0 - \min\left\{\frac{64\epsilon^{1.5}}{\sqrt{\rho}}, \frac{16\epsilon L}{\rho}\right\}.$$

In Case 2, we have

$$\begin{aligned}\ell^{\mathcal{K}} - \ell^0 &\leq -\frac{\theta}{2\eta}\sum_{k=0}^{\mathcal{K}-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2 \\ &\leq -\frac{\theta B^2}{2\eta\mathcal{K}} \leq -\frac{\theta B^2}{2\eta K} = -\frac{8\epsilon^{1.5}}{\sqrt{\rho}},\end{aligned}$$

where we use $\mathcal{K}\sum_{t=0}^{\mathcal{K}-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 > B^2$. So the algorithm will terminate in at most $\frac{\triangle_f\sqrt{\rho}}{\epsilon^{3/2}}$ epochs, and each epoch needs at most K gradient and function evaluations. In the last epoch, similar to the proof of Theorem 2.2, we also have $\|\nabla f(\hat{\mathbf{y}})\| \leq O(\epsilon)$. So we have the following theorem.

Theorem C.3. *Suppose that Assumption 2.1 holds. Let $\eta = \frac{1}{4L}$, $B = \sqrt{\frac{\epsilon}{\rho}}$, $\theta = 4\left(\epsilon\rho\eta^2\right)^{1/4}$, $K = \frac{1}{\theta}$, $\gamma = \frac{\theta^2}{\eta}$, $s = \frac{\gamma}{4\rho}$. Then Algorithm 4 terminates in at most $\frac{\triangle_f L^{1/2}\rho^{1/4}}{\epsilon^{7/4}}$ gradient and function evaluations and the output satisfies $\|\nabla f(\hat{\mathbf{y}})\| \leq 267\epsilon$, where $\triangle_f = f(\mathbf{x}_{int}) - \min_{\mathbf{x}} f(\mathbf{x})$.*

Our complexity improves over the $\mathcal{O}(\epsilon^{-7/4}\log\frac{1}{\epsilon})$ one given in (Jin et al., 2018) by the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor. Although Jin et al. (2018) focus on finding second-order stationary point, their complexity to find approximate first-order stationary point also has the additional $\mathcal{O}(\log\frac{1}{\epsilon})$ factor, see the reasons discussed in Section 2. Our analysis for Case 3 above does not invoke the analysis for strongly convex AGD, and moreover, it is much simpler. The proof in (Jin et al., 2018), although very novel, is quite involved, especially the spectral analysis of the second-order system. It should be noted that we measure the convergence rate at the average of the iterates. When measuring at the final iterate, which is always used in practice, we should use the proof in (Jin et al., 2018), and we conjecture that the $\mathcal{O}(\log\frac{1}{\epsilon})$ factor in unlikely to cancel.

Now, we prove Theorem C.3

Proof. We only need to prove $\|\nabla f(\hat{\mathbf{y}})\| \leq \mathcal{O}(\epsilon)$ in the last epoch. Denote

$$h(\mathbf{x}) = \left\langle\nabla f(\mathbf{x}^0), \mathbf{x}-\mathbf{x}^0\right\rangle + \frac{1}{2}(\mathbf{x}-\mathbf{x}^0)^T\mathbf{H}(\mathbf{x}-\mathbf{x}^0),$$
$$\delta^k = \nabla f(\mathbf{y}^k) - \nabla h(\mathbf{y}^k).$$

Similar to the deduction in Section 3.2, we have

$$\begin{aligned}&\mathbf{x}^{k+1} = \mathbf{y}^k - \eta\nabla h(\mathbf{y}^k) - \eta\delta^k, \\ &\|\delta^k\| \leq \frac{\rho}{2}\|\mathbf{y}^k-\mathbf{x}^0\|^2 \leq 2\rho B^2,\end{aligned} \tag{18a}$$

where we use

$$\|\mathbf{x}^k-\mathbf{x}^0\|^2 \leq k\sum_{t=0}^{k-1}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2 \leq B^2, \forall k \leq K, \tag{19a}$$
$$\|\mathbf{y}^k-\mathbf{x}^0\| \leq 2B, \forall k \leq K, \tag{19b}$$

in the last epoch. Similar to the proof of Theorem 2.2, we have

$$\begin{aligned}\|\nabla h(\hat{\mathbf{y}})\| &= \left\|\frac{1}{K_2-K_1+1}\sum_{k=K_1}^{K_2}\nabla h(\mathbf{y}^k)\right\| \\ &= \frac{1}{\eta(K_2-K_1+1)}\left\|\sum_{k=K_1}^{K_2}\left(\mathbf{x}^{k+1}-\mathbf{y}^k+\eta\delta^k\right)\right\|,\end{aligned}$$

and

$$\left\|\sum_{k=K_1}^{K_2}\left(\mathbf{x}^{k+1}-\mathbf{y}^k+\eta\delta^k\right)\right\|$$
$$=\left\|\sum_{k=K_1}^{K_2}\left(\mathbf{x}^{k+1}-\mathbf{x}^k-(1-\theta)(\mathbf{x}^k-\mathbf{x}^{k-1})+\eta\delta^k\right)\right\|$$
$$=\left\|\mathbf{x}^{K_2+1}-\mathbf{x}^{K_1}-(1-\theta)(\mathbf{x}^{K_2}-\mathbf{x}^{K_1-1})+\eta\sum_{k=K_1}^{K_2}\delta^k\right\|$$
$$=\Big\|\mathbf{x}^{K_2+1}-\mathbf{x}^{K_2}-\mathbf{x}^{K_1}+\mathbf{x}^{K_1-1}+\theta(\mathbf{x}^{K_2}-\mathbf{x}^{K_1-1}) +\eta\sum_{k=K_1}^{K_2}\delta^k\Big\|$$
$$\leq\|\mathbf{x}^{K_2+1}-\mathbf{x}^{K_2}\|+\|\mathbf{x}^{K_1}-\mathbf{x}^{K_1-1}\|+\theta\|\mathbf{x}^{K_2}-\mathbf{x}^0\| +\theta\|\mathbf{x}^{K_1-1}-\mathbf{x}^0\|+\eta\sum_{k=K_1}^{K_2}\|\delta^k\|.$$

$\square$

From $K_2-K_1+1\geq\frac{K}{3}$, (19a), and (18a), we have

$$\|\nabla h(\hat{\mathbf{y}})\|\leq\frac{3}{\eta K}\|\mathbf{x}^{K_2+1}-\mathbf{x}^{K_2}\| +\frac{3}{\eta K}\|\mathbf{x}^{K_1}-\mathbf{x}^{K_1-1}\|+\frac{6\theta B}{\eta K}+2\rho B^2.$$

On the other hand, from the definitions of K_1 and K_2, we have

$$\|\mathbf{x}^{K_2+1}-\mathbf{x}^{K_2}\|^2 \leq\frac{1}{K-\lfloor 2K/3\rfloor}\sum_{k=\lfloor 2K/3\rfloor}^{K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2 \leq\frac{1}{K-\lfloor 2K/3\rfloor}\sum_{k=0}^{K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\leq\frac{3B^2}{K^2},$$

and

$$\|\mathbf{x}^{K_1}-\mathbf{x}^{K_1-1}\|^2\leq\frac{1}{\lceil\frac{K}{3}\rceil}\sum_{k=1}^{\lceil\frac{K}{3}\rceil}\|\mathbf{x}^k-\mathbf{x}^{k-1}\|^2 \leq\frac{1}{\lceil\frac{K}{3}\rceil}\sum_{k=0}^{K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|^2\leq\frac{3B^2}{K^2}.$$

So we have

$$\|\nabla h(\hat{\mathbf{y}})\|\leq\frac{6\sqrt{3}B}{\eta K^2}+\frac{6\theta B}{\eta K}+2\rho B^2,$$

and

$$\|\nabla f(\hat{\mathbf{y}})\|\leq\|\nabla h(\hat{\mathbf{y}})\|+\|\nabla f(\hat{\mathbf{y}})-\nabla h(\hat{\mathbf{y}})\| \leq\|\nabla h(\hat{\mathbf{y}})\|+\frac{\rho}{2}\|\hat{\mathbf{y}}-\mathbf{x}^0\|^2 \leq\frac{6\sqrt{3}B}{\eta K^2}+\frac{6\theta B}{\eta K}+4\rho B^2\leq 267\epsilon.$$

D. Efficient Implementation of the Average

Given $\mathbf{x}^0,\mathbf{x}^1,\cdots,\mathbf{x}^K$ and $\mathbf{y}^0,\mathbf{y}^1,\cdots,\mathbf{y}^K$ sequentially, we want to find $\hat{\mathbf{y}}=\frac{1}{K_0+1}\sum_{k=0}^{K_0}\mathbf{y}^k$ efficiently, where $K_0=\operatorname{argmin}_{\lfloor\frac{K}{2}\rfloor\leq k\leq K-1}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$. We present the implementation in Algorithm 6.

Algorithm 6 Implementation of the Average

Initialize $S_1=S_2=0$, $K_0=0$
for $k=0,1,\cdots,K-1$ **do**
  if $k\leq\lfloor\frac{K}{2}\rfloor$ **then**
    $S_1=S_1+\mathbf{y}^k$, $K_0=k$
  else
    if $\|\mathbf{x}^{K_0+1}-\mathbf{x}^{K_0}\|<\|\mathbf{x}^{k+1}-\mathbf{x}^k\|$ **then**
      $S_2=S_2+\mathbf{y}^k$
    else
      $S_1=S_1+S_2+\mathbf{y}^k$, $S_2=0$, $K_0=k$
    end if
  end if
end for
Output $\frac{S_1}{K_0+1}$

Similarly, we can also implement the average in Algorithm 4 efficiently.

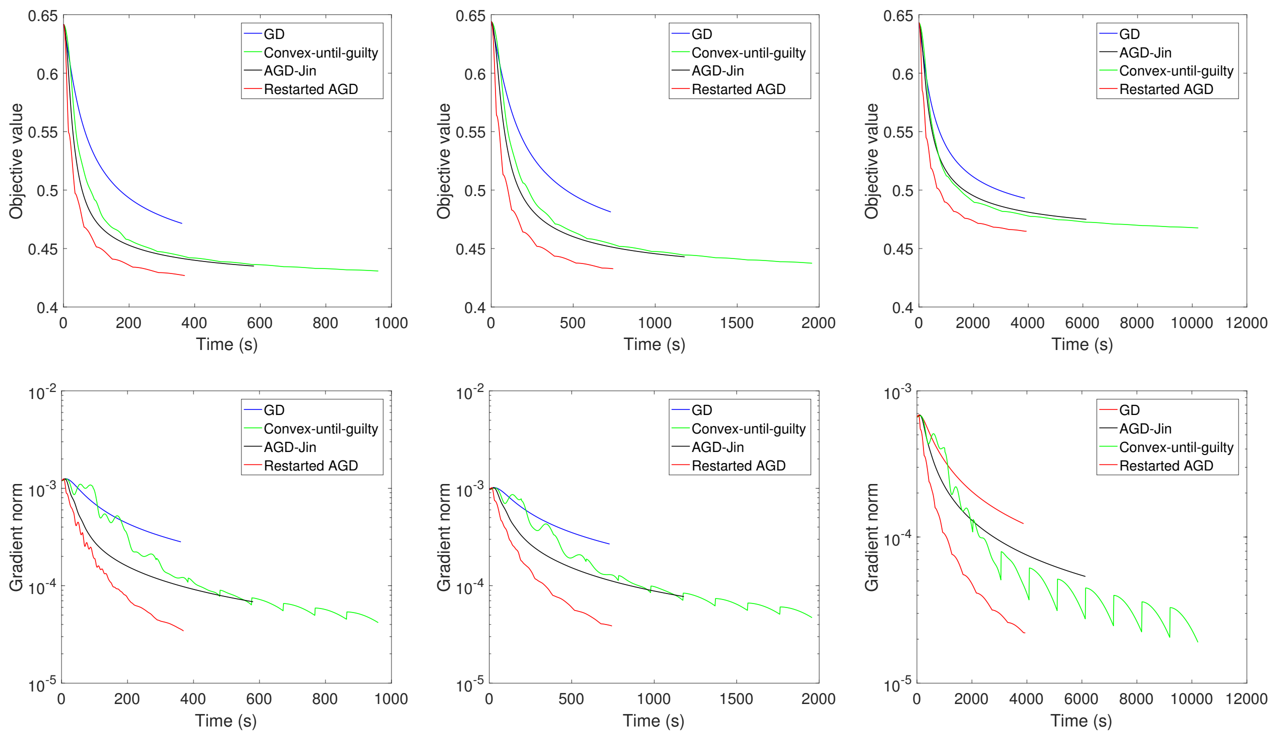
E. Additional Experiments

We consider the one bit matrix completion (Davenport et al., 2014) in this section, where the sign of a random subset of entries is observed, rather than observing the actual entries. Given a probability density function, for example, the logistic function $f(x)=\frac{e^x}{1+e^x}$, we observe the sign of entry $\mathbf{X}_{i,j}$ as $\mathbf{Y}_{i,j}=1$ with probability $f(\mathbf{X}_{i,j})$, and observe the sign as -1 with probability $1-f(\mathbf{X}_{i,j})$. The training model is to minimize the following negative log-likelihood:

$$\min_{\mathbf{X}\in\mathbb{R}^{m\times n}}-\frac{1}{N}\sum_{(i,j)\in\mathcal{O}}\left\{\mathbf{1}_{\mathbf{Y}_{i,j}=1}\log(f(\mathbf{X}_{i,j})) +\mathbf{1}_{\mathbf{Y}_{i,j}=-1}\log(1-f(\mathbf{X}_{i,j}))\right\},$$
$$s.t.\quad \operatorname{rank}(\mathbf{X})\leq r,$$

where $\mathbf{1}_{\mathbf{Y}_{i,j}=1}=\begin{cases}1, & \text{if } \mathbf{Y}_{i,j}=1,\\ 0, & \text{otherwise.}\end{cases}$ We solve the following reformulated matrix factorization model:

$$\min_{\mathbf{U},\mathbf{V}}-\frac{1}{N}\sum_{(i,j)\in\mathcal{O}}\left\{\mathbf{1}_{\mathbf{Y}_{i,j}=1}\log(f((\mathbf{U}\mathbf{V}^T)_{i,j})) +\mathbf{1}_{\mathbf{Y}_{i,j}=-1}\log(1-f((\mathbf{U}\mathbf{V}^T)_{i,j}))\right\} +\frac{1}{2N}\|\mathbf{U}^T\mathbf{U}-\mathbf{V}^T\mathbf{V}\|_F^2,$$



(a) MovieLens-10M (b) MovieLens-20M (c) Netflix

Figure 2. Comparisons of function value and gradient norm on the 1 bit matrix completion problem.

where $\mathbf{U} \in \mathbb{R}^{m \times r}$ and $\mathbf{V} \in \mathbb{R}^{n \times r}$. We compare restarted AGD (Algorithm 2) with Jin's AGD (Jin et al., 2018), the 'convex until guilty' method (Carmon et al., 2017), and gradient descent (GD). The best stepsize is tuned for each method on each data set. We use the same initialization and set the same parameters as those in Section 4, and also run each method for 1000 iterations. Figure 2 plots results. We see that acceleration also takes effect in nonconvex optimization and our restarted AGD performs the best.