REVISITING EXTRA FOR SMOOTH DISTRIBUTED OPTIMIZATION*

HUAN LI† AND ZHOUCHEN LIN‡

Abstract. EXTRA is a popular method for the dencentralized distributed optimization and has broad applications. This paper revisits the EXTRA. Firstly, we give a sharp complexity analysis for EXTRA with the improved $O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ communication and computation complexities for μ-strongly convex and L-smooth problems, where $\sigma_2(W)$ is the second largest singular value of the weight matrix W. When the strong convexity is absent, we prove the $O\left(\left(\frac{L}{\epsilon}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$ complexities. Then, we use the Catalyst framework to accelerate EXTRA and obtain the $O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$ communication and computation complexities for strongly convex and smooth problems and the $O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ complexities for non-strongly convex ones. Our communication complexities of the accelerated EXTRA are only worse by the factors of $\left(\log\frac{L}{\mu(1-\sigma_2(W))}\right)$ and $\left(\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ from the lower complexity bounds for strongly convex and non-strongly convex problems, respectively.

Key words. EXTRA, accelerated distributed optimization, near optimal communication complexity.

AMS subject classifications. 90C25, 90C30

1. Introduction.

In this paper, we consider the following convex problem

$$\min_{x\in\mathbb{R}^n} F(x)=\frac{1}{m}\sum_{i=1}^{m} f_i(x) \tag{1}$$

in the decentralized distributed environment, where m agents form an undirected communication network and collaboratively solve the above problem. Each agent i privately holds a local objective function $f_i(x)$ and can exchange information only with its immediate neighbors. We only consider the network that does not have a centralized agent. Distributed computation has broad applications, ranging from machine learning [8, 10, 2, 1], sensor networks [9], to flow and power control problems [9, 11].

1.1. Literature Review.

Distributed optimization has gained significant attention in engineering applications for a long time [5, 37]. The distributed subgradient method was first proposed in [24] with the convergence and convergence rate analysis for the general network topology, and further extended to the asynchronous variant in [21], the stochastic variant in [29] and a study with fixed step-size in [40]. In [15, 7], the accelerated distributed gradient method in the sense of Nesterov has been proposed and [17] gave a different explanation with sharper analysis, which builds upon the accelerated penalty method. Although the optimal computation complexity and near optimal communication complexity (please see Section 1.4 for the definition of complexity) were proved in [17], the accelerated distributed gradient method employs multiple consensus after each gradient computation and thus places more burdens in the communication-limited environment.

*Submitted to the editors DATE.

Funding: Li is sponsored by Zhejiang Lab (grant no. 2019KB0AB02). Lin is supported by NSF China (grant no.s 61625301 and 61731018), Major Research Project of Zhejiang Lab (grant no.s 2019KB0AC01 and 2019KB0AB02) and Beijing Academy of Artificial Intelligence.

†College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China(lihuanss@nuaa.edu.cn.).

Key Lab. of Machine Perception (MOE), School of EECS, Peking University, Beijing, China. This work was done when Huan Li was a Ph.D student at Peking University.

‡Key Lab. of Machine Perception (MOE), School of EECS, Peking University, Beijing, China (zlin@pku.edu.cn.). Z. Lin is the corresponding author.

A different class of distributed approaches with efficient communication are based on the Lagrangian dual and they work in the dual space. Classical algorithms include dual ascent [36, 30, 38], ADMM [13, 19, 4] and the primal-dual method [16, 31, 12, 14]. Specifically, the accelerated dual ascent [30] and the primal-dual method [31] attain the optimal communication complexities for smooth and nonsmooth problems, respectively. However, the dual based methods require the evaluation of the Fenchel conjugate or the proximal mapping and thus have a larger computation cost per-iteration.

EXTRA [34] and the gradient tracking based method [39, 28] (also named as DIGing in [23]) can be seen as a trade-off between communications and computations, which need equal numbers of communications and gradient computations at each iteration. As a comparison, the accelerated distributed gradient method needs more communications while the dual based methods require more computations. EXTRA uses the differences of gradients and guarantees the convergence to the exact optimal solution with constant step-size. The proximal-gradient variant was studied in [35]. Recently, researchers have established the equivalence between the primal-dual method and EXTRA [12, 20, 14]. Specifically, [12] studied the nonconvex problem, [20] focused on the stochastic optimization while [14] gave a unified framework for EXTRA and the gradient tracking based method. The gradient tracking based method shares some similar features to EXTRA, e.g., using the differences of gradients and constant step-size. The accelerated version of the gradient tracking based method was studied in [27].

In this paper, we revisit EXTRA and give a sharper complexity analysis for the original EXTRA. Then, we propose an accelerated EXTRA, which answers the open problem proposed in [35, Section V] on how to improve the rate of EXTRA with certain acceleration techniques.

1.2. Notation and Assumption. Denote $x_{(i)} \in \mathbb{R}^n$ to be the local copy of the variable x for agent i and $x_{(1:m)}$ to be the set of vectors consisting of $x_{(1)}, \cdots, x_{(m)}$. We introduce the aggregate objective function $f(\mathbf{x})$ of the local variables with its argument $\mathbf{x} \in \mathbb{R}^{m\times n}$ and gradient $\nabla f(\mathbf{x}) \in \mathbb{R}^{m\times n}$ as

$$f(\mathbf{x}) = \sum_{i=1}^{m} f_i(x_{(i)}), \qquad \mathbf{x} = \begin{pmatrix} x_{(1)}^T \\ \vdots \\ x_{(m)}^T \end{pmatrix}, \qquad \nabla f(\mathbf{x}) = \begin{pmatrix} \nabla f_1(x_{(1)})^T \\ \vdots \\ \nabla f_m(x_{(m)})^T \end{pmatrix}. \tag{2}$$

For a given matrix, we use $\|\cdot\|_F$ and $\|\cdot\|_2$ to denote its Frobenius norm and spectral norm, respectively. We denote $\|\cdot\|$ as the l_2 Euclidean norm for a vector. Denote $I \in \mathbb{R}^{m\times m}$ as the identity matrix and $\mathbf{1} = (1, 1, \cdots, 1)^T \in \mathbb{R}^m$ as the vector with all ones. For any matrix $\mathbf{x}$, we denote its average across the rows as

$$\alpha(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^{m} x_{(i)}. \tag{3}$$

Define two operators measuring the consensus violation. The first one is

$$\Pi = I - \frac{1}{m}\mathbf{1}\mathbf{1}^T \in \mathbb{R}^{m\times m} \tag{4}$$

and $\|\Pi\mathbf{x}\|_F$ measures the distance between $x_{(i)}$ and $\alpha(\mathbf{x}), \forall i$. The second one follows [34],

$$U = \sqrt{\frac{I - W}{2}} \in \mathbb{R}^{m\times m}. \tag{5}$$

Let $\mathcal{N}_i$ be the neighbors of agent i and $\mathrm{Span}(U)$ be the linear span of all the columns of U.

We make the following assumptions for the local objectives:

Algorithm 1 EXTRA

Input $F(x)$, K, $x^0_{(1:m)}$, $v^0_{(1:m)}$

for $k = 0, 1, 2, \cdots, K$ **do**

$$x^{k+1}_{(i)} = x^k_{(i)} - \alpha \left(\nabla f_i(x^k_{(i)}) + v^k_{(i)} + \frac{\beta}{2} \left(x^k_{(i)} - \sum_{j \in \mathcal{N}_i} W_{i,j} x^k_{(j)} \right) \right), \forall i.$$

$$v^{k+1}_{(i)} = v^k_{(i)} + \frac{\beta}{2} \left(x^{k+1}_{(i)} - \sum_{j \in \mathcal{N}_i} W_{i,j} x^{k+1}_{(j)} \right), \forall i.$$

end for

Output $x^{K+1}_{(1:m)}$ and $v^{K+1}_{(1:m)}$.

ASSUMPTION 1.

1. *Each $f_i(x)$ is μ-strongly convex: $f_i(y) \geq f_i(x) + \langle \nabla f_i(x), y - x \rangle + \frac{\mu}{2}\|y - x\|^2$. Specially, μ can be zero and we say $f_i(x)$ is convex in this case.*
2. *Each $f_i(x)$ is L-smooth: $f_i(y) \leq f_i(x) + \langle \nabla f_i(x), y - x \rangle + \frac{L}{2}\|y - x\|^2$.*

Then, $F(x)$ and $f(\mathbf{x})$ are also μ-strongly convex and L-smooth. Assume that the set of minimizers of problem (1) is non-empty. Denote x^* as one minimizer and $\mathbf{x}^* = \mathbf{1}(x^*)^T$.

We make the following assumptions for the weight matrix $W \in \mathbb{R}^{m \times m}$ associated to the network:

ASSUMPTION 2.

1. *$W_{i,j} \neq 0$ if and only if agents i and j are neighbors or $i = j$. Otherwise, $W_{i,j} = 0$.*
2. *$W = W^T$, $I \succeq W \succeq -I$ and $W\mathbf{1} = \mathbf{1}$.*
3. *$\sigma_2(W) < 1$, where $\sigma_2(W)$ is the second largest singular value of W.*

Part 2 of Assumption 2 implies that the singular values of W lie in $[0, 1]$ and its largest one $\sigma_1(W)$ equals 1. Moreover, Part 3 can be deduced by part 2 and the assumption that the network is connected. Examples satisfying Assumption 2 can be found in [34].

When minimizing a convex function, the performance of the first-order methods is affected by the smoothness constant L, the strong convexity constant μ, as well as the target accuracy ϵ. When we solve the problem over a network, the connectivity of the network also directly affects the performance. Typically, $\frac{1}{1-\sigma_2(W)}$ is a good indication of the network connectivity [15, 30] and it is often related to m [22, Proposition 5]. For example, for any connected and undirected graph, $\frac{1}{1-\sigma_2(W)} \leq m^2$ [22]. In this paper, we study the complexity of EXTRA with explicit dependence on L, μ, $1 - \sigma_2(W)$ and ϵ.

Denote $x^0_{(1:m)}$ to be the initializers. Assume that $\|x^0_{(i)} - x^*\|^2 \leq R_1$, $\|x^*\|^2 \leq R_1$ and $\|\nabla f_i(x^*)\|^2 \leq R_2, \forall i = 1, \cdots, m$. Then we can simply have

$$\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 \leq mR_1, \quad \|\mathbf{x}^*\|_F^2 \leq mR_1 \quad \text{and} \quad \|\nabla f(\mathbf{x}^*)\|_F^2 \leq mR_2. \tag{6}$$

In this paper, we only regard R_1 and R_2 as the constants which can be dropped in our complexities.

1.3. Proposed Algorithm. Before presenting the proposed algorithm, we first rewrite EXTRA in the primal-dual framework in Algorithm 1. When we set $\alpha = \frac{1}{\beta}$, Algorithm 1 reduces to the original EXTRA [1]. In this paper, we specify $\alpha = \frac{1}{2(L+\beta)}$ and $\beta = L$ for the strongly convex problems to give a faster convergence rate than the original EXTRA, which is crucial to obtain the near optimal communication complexities after acceleration.

We use the Catalyst framework [18] to accelerate Algorithm 1. It has double loops and is described in Algorithm 2. The inner loop calls Algorithm 1 to approximately minimize a

[1] Initialize $\mathbf{v}^0 = \mathbf{0}$ and define $\widetilde{W} = \frac{I+W}{2}$. The second step of Algorithm 1 leads to $\mathbf{v}^k = \beta \sum_{t=1}^k (\widetilde{W} - W)\mathbf{x}^k$. Plugging it into the first step and letting $\alpha = 1/\beta$, it leads to equation (3.5) in [34].

Algorithm 2 Accelerated EXTRA

Initialize $x_{(i)}^0 = y_{(i)}^0$, $v_{(i)}^0 = 0$, $q = \frac{\mu}{\mu+\tau}$, set $\theta_k = \sqrt{q}, \forall k$ if $\mu > 0$, otherwise, set $\theta_0 = 1$ and update $\theta_{k+1} \in (0,1)$ by solving the equation $\theta_{k+1}^2 = (1-\theta_{k+1})\theta_k^2$.
for $k = 0, 1, 2, \cdots, K$ **do**
  Define $g_i^k(x) = f_i(x) + \frac{\tau}{2}\|x - y_{(i)}^k\|^2$ and $G^k(x) = \frac{1}{m}\sum_{i=1}^m g_i^k(x)$.
  $(x_{(1:m)}^{k+1}, v_{(1:m)}^{k+1}) = \text{EXTRA}(G^k(x), T_k, x_{(1:m)}^k, v_{(1:m)}^k)$.
  $y_{(i)}^{k+1} = x_{(i)}^{k+1} + \frac{\theta_k(1-\theta_k)}{\theta_k^2+\theta_{k+1}}(x_{(i)}^{k+1} - x_{(i)}^k), \forall i$.
end for

well-chosen auxiliary function of $G^k(x)$ for T_k iterations with warm-start. T_k and τ are given for two cases:

1. When each $f_i(x)$ is strongly convex with $\mu > 0$, then $\tau = L(1-\sigma_2(W)) - \mu > 0$ and $T_k = O\left(\frac{1}{1-\sigma_2(W)}\log\frac{L}{\mu(1-\sigma_2(W))}\right)$, which is a constant.
2. When each $f_i(x)$ is convex with $\mu = 0$, then $\tau = L(1-\sigma_2(W))$ and $T_k = O\left(\frac{1}{1-\sigma_2(W)}\log\frac{k}{1-\sigma_2(W)}\right)$, which is nearly a constant.

Although Algorithm 2 employs the double loop, it places almost no more burdens than the original EXTRA. A good property of Algorithms 1 and 2 in practice is that they need equal numbers of gradient computations and communications at each iterations.

1.4. Complexities. We study the communication and computation complexities of EXTRA and its accelerated version in this paper. They are presented as the numbers of communications and computations to find an ϵ-optimal solution x such that $F(x) - F(x^*) \leq \epsilon$. We follow [17] to define one communication to be the operation that all the agents receive information from their neighbors once, i.e., $\sum_{j\in\mathcal{N}_i} W_{ij}x_{(j)}$ for all $i = 1, 2, \cdots, m$. One computation is defined to be the gradient evaluations of all the agents once, i.e., $\nabla f_i(x_{(i)})$ for all i. Note that the gradients are evaluated in parallel on each nodes.

To find an ϵ-optimal solution, Algorithm 1 needs $O\left(\left(\frac{L}{\mu} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ and $O\left(\left(\frac{L}{\epsilon} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$ iterations for strongly convex and non-strongly convex problems, respectively. The computation and communication complexities are identical for EXTRA, which equal the number of iterations. For Algorithm 2, we establish the $O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$ complexity for strongly convex problems and the $O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ one for non-strongly convex problems.

Our first contribution is to give a sharp analysis for EXTRA with improved complexity. The complexity of the original EXTRA is at least $O\left(\frac{L^2}{\mu^2(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$[2] for strongly convex problems. For non-strongly convex ones, although the $O\left(\frac{1}{\epsilon}\right)$ complexity was studied in [34], no explicit dependence on $1-\sigma_2(W)$ was given[3]. It is remarkable that the sum of $\frac{L}{\mu}$ (or $\frac{L}{\epsilon}$) and $\frac{1}{1-\sigma_2(W)}$, rather than their product, dominates our complexities. Recall that the full batch centralized gradient descent (GD) performed by a single node need $O\left(\frac{L}{\mu}\log\frac{1}{\epsilon}\right)$ and

[2] [34] did not give an explicit complexity. We try to simplify equation (3.38) in [34] and find it at least $O\left(\frac{L^2}{\mu^2(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$. The true complexity may be larger than $O\left(\frac{L^2}{\mu^2(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$.

[3] [34] proved the $O\left(\frac{1}{K}\right)$ rate in the sense of $\frac{1}{K}\sum_{k=1}^K \|U\lambda^k + \nabla f(\mathbf{x}^k)\|_{\widetilde{W}}^2 \leq O\left(\frac{1}{K}\right)$ and $\frac{1}{K}\sum_{k=1}^K \|U\mathbf{x}^k\|_F^2 \leq O\left(\frac{1}{K}\right)$, where $\widetilde{W} = \frac{I+W}{2}$. [34] omitted the dependence on $1-\sigma_2(W)$ in their analysis.

Non-strongly convex case

Methods	Complexity of gradient computation	Complexity of communication
[34]'s result for EXTRA[4]	$O\left(\frac{1}{\epsilon}\right)$ [34]	$O\left(\frac{1}{\epsilon}\right)$ [34]
Our result for EXTRA	$O\left(\left(\frac{L}{\epsilon}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$	$O\left(\left(\frac{L}{\epsilon}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$
Accelerated Dual Ascent	$O\left(\frac{L}{\epsilon\sqrt{1-\sigma_2(W)}}\log^2\frac{1}{\epsilon}\right)$ [38]	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [38]
Accelerated Penalty Method	$O\left(\sqrt{\frac{L}{\epsilon}}\right)$ [17]	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [17]
Our Accelerated EXTRA	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$
Lower Bound	$O\left(\sqrt{\frac{L}{\epsilon}}\right)$ [25]	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\right)$ [32]

Strongly convex case

Methods	Complexity of gradient computation	Complexity of communication
[34]'s result for EXTRA	at least $O\left(\frac{L^2}{\mu^2(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$ [34]	at least $O\left(\frac{L^2}{\mu^2(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$ [34]
Our result for EXTRA	$O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$	$O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$
Accelerated Dual Ascent	$O\left(\frac{L}{\mu\sqrt{1-\sigma_2(W)}}\log^2\frac{1}{\epsilon}\right)$ [38]	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [30, 38]
Accelerated Penalty Method	$O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ [17]	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log^2\frac{1}{\epsilon}\right)$ [17]
Our Accelerated EXTRA	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$
Lower Bound	$O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ [25]	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [30]

TABLE 1

Complexity comparisons between the accelerated dual ascent, accelerated penalty method with consensus, EXTRA and accelerated EXTRA for smooth convex problems.

$O\left(\frac{L}{\epsilon}\right)$ iterations to find an ϵ-optimal solution for strongly convex and non-strongly convex problems, respectively [25]. If $\frac{1}{1-\sigma_2(W)}$ is smaller than $\frac{L}{\mu}$ (or $\frac{L}{\epsilon}$), we can see that the gradient computation time of EXTRA is nearly $1/m$ that of the full batch centralized GD since EXTRA computes m individual gradients in parallel while the full batch centralized GD computes them sequentially. Thus, EXTRA achieves a linear speed up if we ignore the logarithm factor.

Our second contribution is to give an accelerated EXTRA with the near optimal communication complexity and a competitive computation complexity. In Table 1, we summarize the comparisons to the state-of-art decentralized optimization algorithms, namely, the accelerated dual ascent and accelerated penalty method with consensus. We also present the complexities of the non-accelerated EXTRA and the lower complexity bounds. Our communication complexities of the accelerated EXTRA match the lower bounds except the extra factors of $\left(\log\frac{L}{\mu(1-\sigma_2(W))}\right)$ and $\left(\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$ for strongly convex and non-strongly convex problems, respectively. When high precision is required, i.e., $\frac{1}{\epsilon}>\frac{L}{\mu(1-\sigma_2(W))}$ for strongly convex problems and $\frac{1}{\epsilon}>\frac{1}{1-\sigma_2(W)}$ for non-strongly convex problems, our communication complexities are competitive to the state-of-art ones in [30, 38, 17]. On the other hand, our computation complexities are better than [38] for applications with large $\frac{L}{\epsilon}$ and $\frac{L}{\mu}$ and moderate $\log\frac{1}{1-\sigma_2(W)}$, but worse than [17][5]. Our result is a significant complement to the existing work in the sense that EXTRA and its accelerated version have equal numbers of communications and computations, while the accelerated dual ascent has more computations than communications and the accelerated penalty method needs more communications than computations.

[4] [34] did not give a explicit dependence on $\frac{1}{1-\sigma_2(W)}$.

[5] Although [30] also gives the $O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ computation complexity, [30] defines one computation to be the cost of solving an 'argmin' subproblem. We cite [38] in Table 1, who studies the computation complexity with the total number of gradient computations, which is a more reasonable measurement.

1.5. Paper Organization. The rest of the paper is organized as follows. Section 2 gives a sharper analysis on the original EXTRA and Section 3 develops the accelerated EXTRA. Section 4 proves the complexities and Section 5 gives some numerical experiments. Finally, we conclude in Section 6.

2. Enhanced Results on EXTRA. We give a sharper analysis on EXTRA in this section. Specifically, section 2.1 studies the strongly convex problems and section 2.2 studies the non-strongly convex ones, respectively.

2.1. Sharper Analysis on EXTRA for Strongly Convex Problems. We first describe EXTRA in the primal-dual form. From Assumption 2 and the definition in (5), we know that $\mathbf{x}$ is consensus if and only if $U\mathbf{x} = \mathbf{0}$. Thus, we can reformulate problem (1) as the following linearly constrained problem:

$$\min_{\mathbf{x}\in\mathbb{R}^{m\times n}} f(\mathbf{x}), \quad s.t. \quad U\mathbf{x} = \mathbf{0}. \tag{7}$$

Introduce the augmented Lagrangian function

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \langle \lambda, U\mathbf{x}\rangle + \frac{\beta}{2}\|U\mathbf{x}\|_F^2.$$

Problem (7) can be solved by the classical primal-dual method [12, 14]. Specifically, it uses the Gauss-Seidel-like order to compute the saddle point of the augmented Lagrangian function and consists of the following iterations:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{1}{2(L+\beta)}(\nabla f(\mathbf{x}^k) + U\lambda^k + \beta U^2\mathbf{x}^k), \tag{8a}$$

$$\lambda^{k+1} = \lambda^k + \beta U\mathbf{x}^{k+1}, \tag{8b}$$

where we specify the step-size in the primal step as $\frac{1}{2(L+\beta)}$. Step (8b) involves the operation of $U\mathbf{x}$, which is uncomputable in the distributed environment. We introduce the auxiliary variable

$$\mathbf{v}^k = U\lambda^k.$$

Multiplying both sides of (8b) by U, it leads to

$$\mathbf{v}^{k+1} = \mathbf{v}^k + \beta U^2\mathbf{x}^{k+1}.$$

From the definition of U in (5), we have Algorithm 1. Now, we establish the convergence of Algorithm 1. Define

$$\rho_k = (L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^k - \lambda^*\|_F^2, \tag{11}$$

where $(\mathbf{x}^*, \lambda^*)$ is a KKT point of the saddle point problem $\min_{\mathbf{x}}\max_{\lambda} f(\mathbf{x}) + \langle\lambda, U\mathbf{x}\rangle$. We prove the exponentially diminishing of ρ_k in the following theorem. Specially, we choose a smaller β, i.e., a larger step-size α in the primal step than [34] to obtain a faster convergence rate. More precisely, the original EXTRA uses the step-size of $O\left(\frac{\mu}{L^2}\right)$ [34, Remark 4] and an open problem was proposed in [34] on how to prove linear convergence under the larger step-size of $O\left(\frac{1}{L}\right)$. Our analysis addresses this open problem. We leave the proof in Section 4.1 and describe the crucial tricks there.

THEOREM 1. *Suppose that Assumptions 1 and 2 hold with $\mu > 0$. Let $\mathbf{v}^0 \in Span(U^2)$, $\alpha = \frac{1}{2(L+\beta)}$ and $\beta = L$. Then, for Algorithm 1, we have*

$$\rho_{k+1} \leq (1-\delta)\,\rho_k,$$

where $\delta = \frac{1}{39\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)}$.

Based on Theorem 1, we can give the following corollary, which proves that Algorithm 1 needs $O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{L}{\epsilon(1-\sigma_2(W))}\right)$ iterations to find an ϵ-optimal solution. Recall that $\alpha(\mathbf{x})$ is the average of $x_{(1)},\cdots,x_{(m)}$ defined in (3).

COROLLARY 2. *Under the assumptions of Theorem 1 and let $\mathbf{v}^0 = \mathbf{0}$, Algorithm 1 needs*

$$O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{LR_1+R_2/L}{\epsilon(1-\sigma_2(W))}\right)$$

iterations to achieve an ϵ-optimal solution $\mathbf{x}$ such that

$$F(\alpha(\mathbf{x}))-F(x^*)\leq\epsilon \quad \textit{and} \quad \frac{1}{m}\sum_{i=1}^{m}\left\|x_{(i)}-\alpha(\mathbf{x})\right\|^2\leq\epsilon^2.$$

2.2. Sharper Analysis on EXTRA for Non-strongly Convex Problems. We study EXTRA for non-strongly convex problems in this section. Specifically, we study the original EXTRA in Section 2.2.1 and the regularized EXTRA in Section 2.2.2, respectively.

2.2.1. Complexity for the Original EXTRA. The $O\left(\frac{1}{K}\right)$ convergence rate of EXTRA was well studied in [34, 35]. However, [34, 35] did not establish the explicit dependence on $1-\sigma_2(W)$. In this section, we study the original EXTRA and give the $O\left(\frac{L}{K\sqrt{1-\sigma_2(W)}}\right)$ convergence rate in the following lemma.

LEMMA 3. *Suppose that Assumptions 1 and 2 hold with $\mu = 0$. Let $\alpha = \frac{1}{2(L+\beta)}$ and $\beta = \frac{L}{\sqrt{1-\sigma_2(W)}}$ and define $\hat{\mathbf{x}}^K = \frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k$. Assume that $K \geq \frac{1}{\sqrt{1-\sigma_2(W)}}$. Then, for Algorithm 1, we have*

$$\begin{aligned}
F(\alpha(\hat{\mathbf{x}}^K))-F(x^*) &\leq \frac{34}{K\sqrt{1-\sigma_2(W)}}\left(LR_1+\frac{R_2}{L}\right),\\
\frac{1}{m}\sum_{i=1}^{m}\left\|\hat{x}^K_{(i)}-\alpha(\hat{\mathbf{x}}^K)\right\|^2 &\leq \frac{16}{K^2(1-\sigma_2(W))}\left(R_1+\frac{R_2}{L^2}\right).
\end{aligned}$$

We assume $K \geq \frac{1}{\sqrt{1-\sigma_2(W)}}$ in Lemma 3 and it is a reasonable assumption. Take the linear network as an example, where all the agents connect in a line. For this special network we know $\frac{1}{1-\sigma_2(W)} = m^2$ [22]. Algorithm 1 needs at least m iterations to exchange message between the two farthest nodes in the network. Thus, any convergent method needs at least $\frac{1}{\sqrt{1-\sigma_2(W)}}$ iterations.

In Section 2.1, we establish the $O\left(\left(\frac{L}{\mu}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{L}{\epsilon(1-\sigma_2(W))}\right)$ complexity for strongly convex problems. Naturally, one may expect the $O\left(\frac{L}{\epsilon}+\frac{1}{1-\sigma_2(W)}\right)$ complexity for non-strongly convex ones. However, Lemma 3 only proves the $O\left(\frac{L}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$ complexity. We describe the technical challenges in Section 4.2. It is unclear how to establish the faster rate for the original EXTRA currently and we leave it as an open problem. In the following section, we improve the complexity via solving a regularized problem.

2.2.2. Complexity for the Regularized EXTRA. When the complexity for the strongly convex problems is well studied, the regularization technique is a common way to solve the non-strongly convex ones [3]. Namely, we add a small strongly convex regularizer to the objective and solve the regularized problem instead. Define the regularized version of $F(x)$ as

$$F_\epsilon(x) = \frac{1}{m}\sum_{i=1}^m f_i(x) + \frac{\epsilon}{2}\|x\|^2 \tag{12}$$

and denote $x_\epsilon^* = \operatorname{argmin}_x F_\epsilon(x)$. It can be easily checked that the precisions between problems (1) and (12) satisfies

$$F(x) - F(x^*) \leq F_\epsilon(x) - F_\epsilon(x_\epsilon^*) + \frac{\epsilon}{2}\|x^*\|^2. \tag{13}$$

Thus, to attain an ϵ-optimal solution of problem (1), we only need to find an ϵ-optimal solution of problem (12). Denote $L_\epsilon = L + \epsilon$. Define

$$f_\epsilon(\mathbf{x}) = f(\mathbf{x}) + \frac{\epsilon}{2}\|\mathbf{x}\|_F^2$$

and it is L_ϵ-smooth and ϵ-strongly convex. Problem (12) can be reformulated as the following constrained problem

$$\min_{\mathbf{x}} f_\epsilon(\mathbf{x}), \quad s.t. \quad U\mathbf{x} = 0. \tag{14}$$

Denote $(\mathbf{x}_\epsilon^*, \lambda_\epsilon^*)$ to be a pair of KKT point of problem (14). We can use Algorithm 1 to solve problem (14) and Corollary 2 states that it needs $O\left(\left(\frac{L}{\epsilon} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L}{\epsilon(1-\sigma_2(W))}\right)$ iterations to find an ϵ-optimal solution of problem (12), which is also an ϵ-optimal solution of problem (1).

When $\epsilon \leq 1-\sigma_2(W)$, the complexity of the above regularized EXTRA is dominated by $O\left(\frac{L}{\epsilon}\log\frac{L}{\epsilon}\right)$. We want to further reduce the complexity by the $\left(\log\frac{L}{\epsilon}\right)$ factor. As discussed in Section 4.2, the main reason for the slow rate of the original EXTRA discussed in Section 2.2.1 is that $(L+\beta)\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^0 - \lambda^*\|_F^2$ has the same order of magnitude as $O(1)$, rather than $O(1-\sigma_2(W))$. Our motivation is that we may find a good enough initializer in reasonable time such that $(L+\beta)\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^0 - \lambda^*\|_F^2$ is of the order $O(1-\sigma_2(W))$. With this inspiration, our algorithm consists of two stage. In the first stage, we run Algorithm 1 for K_0 iterations to solve problem (12) and use its output $(\mathbf{x}^{K_0}, \lambda^{K_0})$ as the initializer of the second stage. In the second stage, we run Algorithm 1 on problem (12) again for K iterations and output the averaged solution $\hat{\mathbf{x}}^K$. Although we analyze the method in two stage, we implement it in a single loop and only average over the last K iterations.

The complexity of our two stage regularized EXTRA is described in the following lemma. We can see that the complexity is improved from $O\left(\left(\frac{L}{\epsilon} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L}{\epsilon(1-\sigma_2(W))}\right)$ to $O\left(\left(\frac{L}{\epsilon} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$ via the two stage strategy.

LEMMA 4. *Suppose that Assumptions 1 and 2 hold with $\mu = 0$. Let $\mathbf{v}^0 \in Span(U^2)$, $\alpha = \frac{1}{2(L_\epsilon+\beta)}$ and $\beta = L_\epsilon$. Run Algorithm 1 on problem (12). Then, we only need*

$$O\left(\left(\frac{L}{\epsilon} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$$

iterations for the first stage and $K = O\left(\frac{LR_1 + R_2/L}{\epsilon}\right)$ iterations for the second stage such that

$$F_\epsilon(\alpha(\hat{\mathbf{x}}^K)) - F_\epsilon(x_\epsilon^*) \leq \epsilon \quad \textit{and} \quad \frac{1}{m}\sum_{i=1}^m \left\|\hat{x}_{(i)}^K - \alpha(\hat{\mathbf{x}}^K)\right\|^2 \leq \epsilon^2,$$

where $\hat{\mathbf{x}}^K = \frac{1}{K}\sum_{k=1}^K \mathbf{x}^k$ *in the second stage.*

3. Accelerated EXTRA.

We first review Catalyst and then use it to accelerate EXTRA.

3.1. Catalyst.

Catalyst [18] is a general scheme for accelerating gradient-based optimization methods in the sense of Nesterov. It builds upon the inexact accelerated proximal point algorithm, which consists of the following iterations:

$$x^{k+1} \approx \operatorname*{argmin}_{x\in\mathbb{R}^n} F(x) + \frac{\tau}{2}\|x - y^k\|^2, \tag{15a}$$

$$y^{k+1} = x^{k+1} + \frac{\theta_k(1-\theta_k)}{\theta_k^2 + \theta_{k+1}}(x^{k+1} - x^k), \tag{15b}$$

where θ_k is defined in Algorithm 2. Catalyst employs double loop and approximately solves a sequence of well-chosen auxiliary problems in step (15a) in the inner loop. The following theorem describes the convergence rate for the outer loop.

THEOREM 5. *[33, 18] Suppose that $F(x)$ is convex and the following criterion holds for all $k \le K$ with $\varepsilon_k \le \frac{1}{k^{4+2\xi}}$:*

$$F(x^{k+1}) + \frac{\tau}{2}\|x^{k+1} - y^k\|^2 \le \min_x \left(F(x) + \frac{\tau}{2}\|x - y^k\|^2\right) + \varepsilon_k, \tag{16}$$

where ξ can be any small positive constant. Then, Catalyst generates iterates $(x^k)_{k=0}^{K+1}$ such that

$$F(x^{K+1}) - F(x^*) \le \frac{1}{(K+2)^2}\left(6\tau\|x^0 - x^*\|^2 + \frac{48}{\xi^2} + \frac{12}{1+2\xi}\right).$$

Suppose that $F(x)$ is μ-strongly convex and (16) holds for all $k \le K$ with the precision of $\varepsilon_k \le \frac{2(F(x^0)-F(x^*))}{9}(1-\rho)^{k+1}$, where $\rho < \sqrt{q}$ and $q = \frac{\mu}{\mu+\tau}$. Then, Catalyst generates iterates $(x_k)_{k=0}^{K+1}$ such that

$$F(x^{K+1}) - F(x^*) \le \frac{8}{(\sqrt{q}-\rho)^2}(1-\rho)^{K+2}(F(x^0) - F(x^*)).$$

Briefly, Catalyst uses some linearly convergent method to solve the subproblem in step (15a) with warm-start, balances the outer loop and inner loop and attains the near optimal global complexities.

3.2. Accelerating EXTRA via Catalyst.

We first establish the relation between Algorithm 2 and Catalyst. Recall the definition of $G^k(x)$ in Algorithm 2:

$$G^k(x) = \frac{1}{m}\sum_{i=1}^m g_i^k(x), \quad \text{where} \quad g_i^k(x) = f_i(x) + \frac{\tau}{2}\|x - y_{(i)}^k\|^2,$$

which is $(L+\tau)$-smooth and $(\mu+\tau)$-strongly convex. Denote $L_g = L + \tau$ and $\mu_g = \mu + \tau$ for simplicity. We can easily check that

$$\begin{aligned} G^k(x) =& \frac{1}{m}\sum_{i=1}^m f_i(x) + \frac{\tau}{2}\sum_{i=1}^m \frac{1}{m}\|x - y_{(i)}^k\|^2 \\ =& \frac{1}{m}\sum_{i=1}^m f_i(x) + \frac{\tau}{2}\|x - \alpha(\mathbf{y}^k)\|^2 - \frac{\tau}{2}\|\alpha(\mathbf{y}^k)\|^2 + \frac{\tau}{2}\sum_{i=1}^m \frac{1}{m}\|y_{(i)}^k\|^2. \end{aligned}$$

Recall that $\alpha(\mathbf{y}^k)$ is the average of $y_{(1)}^k, \cdots, y_{(m)}^k$ defined in (3). In Algorithm 2, we call EXTRA to minimize $G^k(x)$ approximately, i.e., to minimize $F(x) + \frac{\tau}{2}\|x - \alpha(\mathbf{y}^k)\|^2$. Thus, Algorithm 2 can be interpreted as

$$\begin{aligned}\alpha(\mathbf{x}^{k+1}) &\approx \operatorname*{argmin}_x F(x) + \frac{\tau}{2}\|x - \alpha(\mathbf{y}^k)\|^2,\\ \alpha(\mathbf{y}^{k+1}) &= \alpha(\mathbf{x}^{k+1}) + \frac{\theta_k(1-\theta_k)}{\theta_k^2 + \theta_{k+1}}(\alpha(\mathbf{x}^{k+1}) - \alpha(\mathbf{x}^k)),\end{aligned}$$

and it belongs to the Catalyst framework. Thus, we only need to ensure (16), i.e., $G^k(\alpha(\mathbf{x}^{k+1})) \leq \min_x G^k(x) + \varepsilon_k$ for all k. Catalyst requires the liner convergence in the form of

$$G^k(z^t) - \min_x G^k(x) \leq (1-\delta)^t \left(G^k(z^0) - \min_x G^k(x)\right)$$

when solving the subproblem in step (15a), which is not satisfied for Algorithm 1 due to the existence of terms $\|\lambda^k - \lambda^*\|_F^2$ and $\|\lambda^{k+1} - \lambda^*\|_F^2$ in Theorem 1. Thus, the conclusion in [18] cannot be directly applied to Algorithm 2. By analyzing the inner-loop carefully, we can have the following theorem, which establishes that a suitable constant setup of T_k is sufficient to ensure (16) in the strongly convex case and thus allows us to use the Catalyst framework to distributed optimization, where T_k is the number of inner iterations when calling Algorithm 1.

THEOREM 6. *Suppose that Assumptions 1 and 2 hold with $\mu > 0$. We only need to set $T_k = O\left(\left(\frac{L+\tau}{\mu+\tau} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L+\tau}{\mu(1-\sigma_2(W))}\right)$ in Algorithm 2 such that $G^k(\alpha(\mathbf{x}^{k+1})) \leq \min_x G^k(x) + \varepsilon_k$ holds for all k, where ε_k is defined in Theorem 5.*

Based on Theorems 5 and 6, we can establish the global complexity via finding the optimal balance between the inner-loop and outer-loop. Specifically, the total number of inner iterations is

$$\sum_{k=0}^{\sqrt{1+\frac{\tau}{\mu}}\log\frac{1}{\epsilon}} T_k = \sqrt{1+\frac{\tau}{\mu}}\left(\frac{L+\tau}{\mu+\tau} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L+\tau}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}.$$

We obtain the minimal value with the optimal setting of τ, which is described in the following corollary. On the other hand, when we set $\tau \approx 0$, it approximates the original EXTRA.

COROLLARY 7. *Under the settings of Theorem 6 and letting $\tau = L(1-\sigma_2(W)) - \mu$, Algorithm 2 needs*

$$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\right)$$

total inner iterations to achieve an ϵ-optimal solution such that

$$F(\alpha(\mathbf{x})) - F(x^*) \leq \epsilon \quad \textit{and} \quad \frac{1}{m}\sum_{i=1}^m \left\|x_{(i)} - \alpha(\mathbf{x})\right\|^2 \leq \epsilon^2.$$

When the strong-convexity is absent, we have the following conclusions, which are the counterparts of Theorem 6 and Corollary 7.

THEOREM 8. *Suppose that Assumptions 1 and 2 hold with $\mu = 0$. We only need to set $T_k = O\left(\left(\frac{L+\tau}{\tau} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{k}{1-\sigma_2(W)}\right)$ in Algorithm 2 such that $G^k(\alpha(\mathbf{x}^{k+1})) \leq \min_x G^k(x) + \varepsilon_k$ holds for all k.*

COROLLARY 9. *Under the settings of Theorem 8 and letting* $\tau = L(1-\sigma_2(W))$*, Algorithm 2 needs*

$$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}\right)$$

total inner iterations to achieve an ϵ*-optimal solution such that*

$$F(\alpha(\mathbf{x}))-F(x^*)\leq\epsilon \quad \textit{and} \quad \frac{1}{m}\sum_{i=1}^{m}\left\|x_{(i)}-\alpha(\mathbf{x})\right\|^2\leq\epsilon^2.$$

The accelerated EXTRA needs to know $\frac{1}{1-\sigma_2(W)}$ in advance to set T_k and τ. Generally speaking, $\frac{1}{1-\sigma_2(W)}$ relates to the global connectivity of the network. [22, Proposition 5] gives the estimation of $\frac{1}{1-\sigma_2(W)}$ by m for many frequently-used networks, e.g., the 2-D graph, geometric graph, expander graph and the Erdős-Rényi random graph. Please see [22] for the details.

4. Proof of Theorems. In this section, we give the proofs of the theorems, corollaries and lemmas in Sections 2 and 3. We first present several supporting lemmas, which will be used in our analysis.

LEMMA 10. *Assume that Assumption 2 holds. Then, we have* $\|\Pi\mathbf{x}\|_F\leq\sqrt{\frac{2}{1-\sigma_2(W)}}\|U\mathbf{x}\|_F$.

The proof is similar to that of [17, Lemma 4] and we omit the details.

LEMMA 11. *Suppose that* $\mathbf{x}^*$ *is the optimal solution of problem (7). There exists* $\lambda^*\in$ *Span*(U) *such that* $(\mathbf{x}^*,\lambda^*)$ *is a KKT point of the saddle point problem* $\min_{\mathbf{x}}\max_{\lambda}f(\mathbf{x})+\langle\lambda,U\mathbf{x}\rangle$*. For* λ^* *and any* $\lambda\in$ *Span*(U)*, we have* $\|\lambda^*\|_F\leq\frac{\sqrt{2}\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}$ *and* $\|U(\lambda-\lambda^*)\|_F^2\geq\frac{1-\sigma_2(W)}{2}\|\lambda-\lambda^*\|_F^2$.

The existence of $\lambda^*\in\text{Span}(U)$ was proved in [34, Lemma 3.1] and $\|\lambda^*\|_F\leq\frac{\|\nabla f(\mathbf{x}^*)\|_F}{\widetilde{\sigma}_{\min}(U)}$ was proved in [16, Theorem 2], where $\widetilde{\sigma}_{\min}(U)$ denotes the smallest nonzero singular value of U and it is equal to $\sqrt{\frac{1-\sigma_2(W)}{2}}$. The last inequality can be obtained from a similar induction to the proof of Lemma 10 and we omit the details. From Lemma 11 we can see that when we study the complexities on the dependence of $1-\sigma_2(W)$, we should deal with $\|\lambda^*\|_F$ carefully. $\|\lambda^*\|_F$ cannot be regarded as a constant that can be dropped in the complexities.

LEMMA 12. *Assume that* $f(\mathbf{x})$ *is* μ*-strongly convex and* L*-smooth. Then, we have*

$$\frac{\mu}{2}\|\mathbf{x}-\mathbf{x}^*\|_F^2\leq f(\mathbf{x})-f(\mathbf{x}^*)+\langle\lambda^*,U\mathbf{x}\rangle\leq\frac{L}{2}\|\mathbf{x}-\mathbf{x}^*\|_F^2. \tag{17}$$

Assume that $f(\mathbf{x})$ *is convex and* L*-smooth. Then, we have*

$$\frac{1}{2L}\|\nabla f(\mathbf{x})+U\lambda^*\|_F^2\leq f(\mathbf{x})-f(\mathbf{x}^*)+\langle\lambda^*,U\mathbf{x}\rangle\,. \tag{18}$$

Proof: We can easily see that $f(\mathbf{x})+\langle\lambda^*,U\mathbf{x}\rangle$ is μ-strongly convex and L-smooth in $\mathbf{x}$. Since $\mathbf{x}^*=\text{argmin}_{\mathbf{x}}f(\mathbf{x})+\langle\lambda^*,U\mathbf{x}\rangle$ and $U\mathbf{x}^*=\mathbf{0}$, we have (17). From $\nabla f(\mathbf{x}^*)+U\lambda^*=\mathbf{0}$ and the smoothness of $f(\mathbf{x})+\langle\lambda^*,U\mathbf{x}\rangle$ [26, Theorem 2.1.5], we have (18). □

LEMMA 13. *Suppose that Assumptions 1 and 2 hold with* $\mu=0$*. Assume that* $f(\mathbf{x})-f(\mathbf{x}^*)+\langle\lambda^*,U\mathbf{x}\rangle\leq\epsilon_1$ *and* $\|U\mathbf{x}\|_F\leq\epsilon_2$*. Then, we have*

$$F(\alpha(\mathbf{x}))-F(x^*)\leq\frac{1}{m}\left(\epsilon_1+\frac{3\|\nabla f(\mathbf{x}^*)\|_F+2L\|\mathbf{x}-\mathbf{x}^*\|_F}{\sqrt{1-\sigma_2(W)}}\epsilon_2+\frac{L}{1-\sigma_2(W)}\epsilon_2^2\right).$$

Proof: Recall that $\frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x} = \mathbf{1}(\alpha(\mathbf{x}))^T$ from (3) and $\mathbf{x}^* = \mathbf{1}(x^*)^T$. From the definitions of $F(x)$ and $f(\mathbf{x})$ in (1) and (2), respectively, we have $F(\alpha(\mathbf{x})) = \frac{1}{m}f\left(\frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}\right)$ and $F(x^*) = \frac{1}{m}f(\mathbf{x}^*)$. Thus, we only need to bound $f\left(\frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}\right) - f(\mathbf{x}^*)$.

$$
\begin{aligned}
&f\left(\frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}\right) - f(\mathbf{x}^*)\\
=&f\left(\frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}\right) - f(\mathbf{x}) + f(\mathbf{x}) - f(\mathbf{x}^*)\\
\overset{a}{\leq}&\left\langle \nabla f(\mathbf{x}), \frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x} - \mathbf{x}\right\rangle + \frac{L}{2}\|\Pi\mathbf{x}\|_F^2 + f(\mathbf{x}) - f(\mathbf{x}^*)\\
\overset{b}{\leq}&\left(\|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{x} - \mathbf{x}^*\|_F\right)\|\Pi\mathbf{x}\|_F + \frac{L}{2}\|\Pi\mathbf{x}\|_F^2 + f(\mathbf{x}) - f(\mathbf{x}^*)\\
\overset{c}{\leq}&\left(\|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{x} - \mathbf{x}^*\|_F\right)\sqrt{\frac{2}{1-\sigma_2(W)}}\|U\mathbf{x}\|_F + \frac{L}{1-\sigma_2(W)}\|U\mathbf{x}\|_F^2\\
&+ f(\mathbf{x}) - f(\mathbf{x}^*) + \langle\lambda^*, U\mathbf{x}\rangle + \|\lambda^*\|_F\|U\mathbf{x}\|_F\\
\overset{d}{\leq}&\left(\|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{x} - \mathbf{x}^*\|_F\right)\sqrt{\frac{2}{1-\sigma_2(W)}}\|U\mathbf{x}\|_F + \frac{L}{1-\sigma_2(W)}\|U\mathbf{x}\|_F^2\\
&+ f(\mathbf{x}) - f(\mathbf{x}^*) + \langle\lambda^*, U\mathbf{x}\rangle + \frac{\sqrt{2}\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}\|U\mathbf{x}\|_F,
\end{aligned}
$$

where we use the smoothness of $f(\mathbf{x})$ and (4) in $\overset{a}{\leq}$ and $\overset{b}{\leq}$, Lemma 10 and $-\langle\lambda^*, U\mathbf{x}\rangle \leq \|\lambda^*\|_F\|U\mathbf{x}\|_F$ in $\overset{c}{\leq}$ and Lemma 11 in $\overset{d}{\leq}$. □

The following lemma is the well-known coerciveness property of the proximal operator.

LEMMA 14. *[18, Lemma 22] Given a convex function $F(x)$ and a positive constant τ, define $p(y) = \operatorname{argmin}_x F(x) + \frac{\tau}{2}\|x - y\|^2$. For any y and y', the following inequality holds,*

$$\|y - y'\| \geq \|p(y) - p(y')\|.$$

At last, we study the regularized problem (14).

LEMMA 15. *Suppose that Assumptions 1 and 2 hold with $\mu = 0$. Then, we have $\|\mathbf{x}^* - \mathbf{x}_\epsilon^*\|_F \leq \|\mathbf{x}^*\|_F$ and $\|\mathbf{x}_\epsilon^*\|_F \leq 2\|\mathbf{x}^*\|_F$.*

The proof is similar to that of [3, Claim 3.4]. We omit the details.

4.1. Proofs of Theorem 1 and Corollary 2. Now, we prove Theorem 1, which is based on the following lemma. It gives a progress in one iteration of Algorithm 1. Some techniques in this proof have already appeared in [34] and we present the proof for the sake of completeness.

LEMMA 16. *Suppose that Assumptions 1 and 2 hold with $\mu = 0$. Then, for procedure (8a)-(8b), we have*

$$
\begin{aligned}
&f(\mathbf{x}^{k+1}) - f(\mathbf{x}^*) + \left\langle\lambda^*, U\mathbf{x}^{k+1}\right\rangle\\
\leq&(L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 - (L+\beta)\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_F^2\\
&+ \frac{1}{2\beta}\|\lambda^k - \lambda^*\|_F^2 - \frac{1}{2\beta}\|\lambda^{k+1} - \lambda^*\|_F^2 - \frac{\beta+L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2.
\end{aligned}
\tag{19}
$$

Proof: From the L-smoothness and convexity of $f(\mathbf{x})$, we have

$$\begin{aligned}
f(\mathbf{x}^{k+1}) \leq & f(\mathbf{x}^k) + \left\langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2 \\
= & f(\mathbf{x}^k) + \left\langle \nabla f(\mathbf{x}^k), \mathbf{x}^* - \mathbf{x}^k \right\rangle + \left\langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^* \right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2 \\
\leq & f(\mathbf{x}^*) + \left\langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^* \right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2.
\end{aligned}$$

Plugging (8a) into the above inequality, adding $\left\langle \lambda^*, U\mathbf{x}^{k+1} \right\rangle$ to both sides and rearranging the terms, we have

$$\begin{aligned}
& f(\mathbf{x}^{k+1}) - f(\mathbf{x}^*) + \left\langle \lambda^*, U\mathbf{x}^{k+1} \right\rangle \\
\leq & - \left\langle 2(L+\beta)(\mathbf{x}^{k+1} - \mathbf{x}^k) + U\lambda^k + \beta U^2 \mathbf{x}^k, \mathbf{x}^{k+1} - \mathbf{x}^* \right\rangle \\
& + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2 + \left\langle \lambda^*, U\mathbf{x}^{k+1} \right\rangle \\
\overset{a}{=} & - 2(L+\beta)\left\langle \mathbf{x}^{k+1} - \mathbf{x}^k, \mathbf{x}^{k+1} - \mathbf{x}^* \right\rangle - \frac{1}{\beta}\left\langle \lambda^k - \lambda^*, \lambda^{k+1} - \lambda^k \right\rangle \\
& - \beta \left\langle U\mathbf{x}^k, U\mathbf{x}^{k+1} \right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2,
\end{aligned}$$

where we use $U\mathbf{x}^* = 0$ and (8b) in $\overset{a}{=}$. Using the identity of $2\langle a, b\rangle = \|a\|^2 + \|b\|^2 - \|a-b\|^2$, we have

$$\begin{aligned}
& f(\mathbf{x}^{k+1}) - f(\mathbf{x}^*) + \left\langle \lambda^*, U\mathbf{x}^{k+1} \right\rangle \\
\leq & (L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 - (L+\beta)\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_F^2 \\
& + \frac{1}{2\beta}\|\lambda^k - \lambda^*\|_F^2 - \frac{1}{2\beta}\|\lambda^{k+1} - \lambda^*\|_F^2 + \frac{1}{2\beta}\|\lambda^{k+1} - \lambda^k\|_F^2 \\
& - \frac{\beta}{2}\|U\mathbf{x}^k\|_F^2 - \frac{\beta}{2}\|U\mathbf{x}^{k+1}\|_F^2 + \frac{\beta}{2}\|U\mathbf{x}^{k+1} - U\mathbf{x}^k\|_F^2 - \left(\frac{L}{2} + \beta\right)\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2 \\
\overset{b}{\leq} & (L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 - (L+\beta)\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_F^2 \\
& + \frac{1}{2\beta}\|\lambda^k - \lambda^*\|_F^2 - \frac{1}{2\beta}\|\lambda^{k+1} - \lambda^*\|_F^2 - \frac{\beta + L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2,
\end{aligned}$$

where we use (8b) and $\|U\|_2^2 \leq 1$ in $\overset{b}{\leq}$. □

A crucial property in (19) is that we keep the term $-\frac{\beta+L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2$, which will be used in the following proof to eliminate term $\left(\frac{1}{\nu} - 1\right)\frac{9(L+\beta)^2}{2L}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_F^2$ to attain (21). In the following proof of Theorem 1, we use the strong-convexity and smoothness of $f(\mathbf{x})$ to obtain two inequalities, i.e., (21) and (22). A convex combination leads to (23). The key thing here is to design the parameters carefully. Otherwise, we may only obtain a suboptimal result with a worse dependence on $\frac{L}{\mu}$ and $\frac{1}{1-\sigma_2(W)}$.

Proof of Theorem 1: We use (18) to upper bound $\|\lambda^{k+1} - \lambda^*\|_F^2$. From procedure (8a)-(8b), we have

$$2(L+\beta)\left(\mathbf{x}^{k+1} - \mathbf{x}^k\right) + \nabla f(\mathbf{x}^k) + U\lambda^{k+1} + \beta U^2(\mathbf{x}^k - \mathbf{x}^{k+1}) = 0.$$

Thus, we obtain

$$
\begin{aligned}
&\frac{1}{2L}\|\nabla f(\mathbf{x}^{k+1})+U\lambda^*\|_F^2\\
=&\frac{1}{2L}\left\|2(L+\beta)\left(\mathbf{x}^{k+1}-\mathbf{x}^k\right)+\beta U^2(\mathbf{x}^k-\mathbf{x}^{k+1})+\nabla f(\mathbf{x}^k)-\nabla f(\mathbf{x}^{k+1})+U(\lambda^{k+1}-\lambda^*)\right\|_F^2\\
\overset{c}{\geq}&\frac{1-\nu}{2L}\|U(\lambda^{k+1}-\lambda^*)\|_F^2\\
&-\frac{1/\nu-1}{2L}\left\|2(L+\beta)\left(\mathbf{x}^{k+1}-\mathbf{x}^k\right)+\beta U^2(\mathbf{x}^k-\mathbf{x}^{k+1})+\nabla f(\mathbf{x}^k)-\nabla f(\mathbf{x}^{k+1})\right\|_F^2\\
\overset{d}{\geq}&\frac{(1-\nu)(1-\sigma_2(W))}{4L}\|\lambda^{k+1}-\lambda^*\|_F^2-\left(\frac{1}{\nu}-1\right)\frac{9(L+\beta)^2}{2L}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|_F^2,
\end{aligned}
\tag{20}
$$

where we use $\|a+b\|^2\geq(1-\nu)\|a\|^2-(1/\nu-1)\|b\|^2$ for some $\nu\in(0,1)$ in $\overset{c}{\geq}$, Lemma 11 and the smoothness of $f(\mathbf{x})$ in $\overset{d}{\geq}$. Lemma 11 requires $\lambda^k\in\text{Span}(U)$. From the initialization and (8b), we know it holds for all k.

Letting $\nu=\frac{9(\beta+L)}{9(\beta+L)+L}$, then $\left(\frac{1}{\nu}-1\right)\frac{9(L+\beta)^2}{2L}=\frac{L+\beta}{2}$. Plugging ν into the above inequality and using (18) and (19), we have

$$
\begin{aligned}
\frac{1-\sigma_2(W)}{36(\beta+L)+4L}\|\lambda^{k+1}-\lambda^*\|_F^2\leq&(L+\beta)\|\mathbf{x}^k-\mathbf{x}^*\|_F^2-(L+\beta)\|\mathbf{x}^{k+1}-\mathbf{x}^*\|_F^2\\
&+\frac{1}{2\beta}\|\lambda^k-\lambda^*\|_F^2-\frac{1}{2\beta}\|\lambda^{k+1}-\lambda^*\|_F^2.
\end{aligned}
\tag{21}
$$

From (17) and (19), we also have

$$
\begin{aligned}
\frac{\mu}{2}\|\mathbf{x}^{k+1}-\mathbf{x}^*\|_F^2\leq&(L+\beta)\|\mathbf{x}^k-\mathbf{x}^*\|_F^2-(L+\beta)\|\mathbf{x}^{k+1}-\mathbf{x}^*\|_F^2\\
&+\frac{1}{2\beta}\|\lambda^k-\lambda^*\|_F^2-\frac{1}{2\beta}\|\lambda^{k+1}-\lambda^*\|_F^2.
\end{aligned}
\tag{22}
$$

Multiplying (21) by η, multiplying (22) by $1-\eta$, adding them together and rearranging the terms, we have

$$
\begin{aligned}
&\left(L+\beta+\frac{(1-\eta)\mu}{2}\right)\|\mathbf{x}^{k+1}-\mathbf{x}^*\|_F^2+\left(\frac{1}{2\beta}+\frac{\eta(1-\sigma_2(W))}{36(\beta+L)+4L}\right)\|\lambda^{k+1}-\lambda^*\|_F^2\\
\leq&(L+\beta)\|\mathbf{x}^k-\mathbf{x}^*\|_F^2+\frac{1}{2\beta}\|\lambda^k-\lambda^*\|_F^2.
\end{aligned}
\tag{23}
$$

Letting $\frac{(1-\eta)\mu}{2(L+\beta)}=\frac{\beta\eta(1-\sigma_2(W))}{18(\beta+L)+2L}$, we have $\eta=\frac{\frac{\mu}{2(L+\beta)}}{\frac{\mu}{2(L+\beta)}+\frac{\beta(1-\sigma_2(W))}{18(\beta+L)+2L}}$. Plugging it into (23) and recalling the definition of ρ_k in (11), it leads to

$$
\left(1+\frac{\mu\beta(1-\sigma_2(W))}{\mu(18(\beta+L)+2L)+2(L+\beta)\beta(1-\sigma_2(W))}\right)\rho_{k+1}\leq\rho_k.
$$

We can easily check

$$
\begin{aligned}
&\frac{\mu\beta(1-\sigma_2(W))}{\mu(18(\beta+L)+2L)+2(L+\beta)\beta(1-\sigma_2(W))}\\
=&\frac{\mu(1-\sigma_2(W))}{\frac{20L\mu}{\beta}+2\beta(1-\sigma_2(W))+2L(1-\sigma_2(W))+18\mu}\\
\overset{e}{\geq}&\frac{1}{38}\frac{\mu(1-\sigma_2(W))}{L(1-\sigma_2(W))+\mu}
\end{aligned}
$$

by letting $\beta = L$ in $\overset{e}{\geq}$. Thus, we have the conclusion. □

At last, we prove Corollary 2.

Proof: From Theorem 1, $\lambda^0 = \mathbf{0}$, (6), $\beta = L$ and Lemma 11, we have

$$\begin{aligned}(L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^k - \lambda^*\|_F^2 &\leq (1-\delta)^k \left((L+\beta)\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^*\|_F^2\right)\\ &\leq (1-\delta)^k \frac{2m(LR_1 + R_2/L)}{1-\sigma_2(W)}.\end{aligned} \tag{24}$$

On the other hand, from $\mathbf{x}^* = \mathbf{1}(x^*)^T$, the definition of $\alpha(\mathbf{x})$ in (3), the convexity of $\|\cdot\|^2$ and the smoothness of $F(x)$, we have

$$\frac{1}{m}\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 = \frac{1}{m}\sum_{i=1}^m \|x_{(i)}^k - x^*\|^2 \geq \|\alpha(\mathbf{x}^k) - x^*\|^2 \geq \frac{2}{L}\left(F(\alpha(\mathbf{x}^k)) - F(x^*)\right). \tag{25}$$

So we have

$$F(\alpha(\mathbf{x}^k)) - F(x^*) \leq (1-\delta)^k \frac{LR_1 + R_2/L}{1-\sigma_2(W)}.$$

On the other hand, since $\frac{1}{m}\sum_{i=1}^m \left\|x_{(i)} - \alpha(\mathbf{x})\right\|^2 = \frac{1}{m}\|\Pi\mathbf{x}\|_F^2$, we only need to bound $\|\Pi\mathbf{x}\|_F^2$.

$$\begin{aligned}\|\Pi\mathbf{x}^k\|_F^2 &\overset{a}{\leq} \frac{2}{1-\sigma_2(W)}\|U\mathbf{x}^k\|_F^2 \overset{b}{=} \frac{2}{(1-\sigma_2(W))\beta^2}\|\lambda^k - \lambda^{k-1}\|_F^2\\ &\leq \frac{4}{(1-\sigma_2(W))\beta^2}\left(\|\lambda^k - \lambda^*\|_F^2 + \|\lambda^{k-1} - \lambda^*\|_F^2\right)\\ &\overset{c}{\leq} \frac{16}{(1-\sigma_2(W))\beta}\left((L+\beta)\|\mathbf{x}^{k-1} - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^{k-1} - \lambda^*\|_F^2\right)\\ &\overset{d}{\leq} (1-\delta)^{k-1}\frac{32m(R_1 + R_2/L^2)}{(1-\sigma_2(W))^2},\end{aligned} \tag{26}$$

where we use Lemma 10 in $\overset{a}{\leq}$, (8b) in $\overset{b}{=}$, $\|\lambda^k - \lambda^*\|_F^2 \leq 2\beta\rho_k \leq 2\beta\rho_{k-1}$ and $\|\lambda^{k-1} - \lambda^*\|_F^2 \leq 2\beta\rho_{k-1}$ in $\overset{c}{\leq}$, (24) and $\beta = L$ in $\overset{d}{\leq}$. The proof completes. □

4.2. Proofs of Lemmas 3 and 4. Lemma 3 only proves the $O\left(\frac{L}{K\sqrt{1-\sigma_2(W)}}\right)$ convergence rate, rather than $O\left(\frac{L}{K}\right)$. In fact, from Lemma 13, to prove the $O\left(\frac{1}{K}\right)$ convergence rate, we should establish $\|U\mathbf{x}\|_F \leq O\left(\frac{\sqrt{m(1-\sigma_2(W))}}{K}\right)$. However, from (29), we know that $\|U\mathbf{x}\|_F$ has only the same order of magnitude as $O\left(\frac{\sqrt{m}}{K}\sqrt{R_1 + \frac{R_2}{\beta^2(1-\sigma_2(W))}}\right)$. We find that $\beta = \frac{L}{\sqrt{1-\sigma_2(W)}}$ is the best choice to balance the terms in (31).

Proof of Lemma 3: Summing (19) over $k = 0, 1, \cdots, K-1$, dividing both sides by K, using the convexity of $f(\mathbf{x})$ and the definition of $\hat{\mathbf{x}}^K$, we have

$$f(\hat{\mathbf{x}}^K) - f(\mathbf{x}^*) + \left\langle \lambda^*, U\hat{\mathbf{x}}^K\right\rangle \leq \frac{1}{K}\left((L+\beta)\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^0 - \lambda^*\|_F^2\right). \tag{27}$$

On the other hand, since $f(\mathbf{x}) - f(\mathbf{x}^*) + \langle\lambda^*, U\mathbf{x}\rangle \geq 0, \forall\mathbf{x}$ from (18), we also have

$$(L+\beta)\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 \leq (L+\beta)\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 + \frac{1}{2\beta}\|\lambda^0 - \lambda^*\|_F^2, \quad \forall k = 1, \cdots, K \tag{28}$$

and

$$\frac{1}{2\beta}\|\lambda^K-\lambda^*\|_F^2 \le (L+\beta)\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{2\beta}\|\lambda^0-\lambda^*\|_F^2$$

from (19). Using (8b) and the definition of $\hat{\mathbf{x}}^K$, we further have

$$\begin{aligned}\|U\hat{\mathbf{x}}^K\|_F^2 =& \frac{1}{\beta^2K^2}\|\lambda^K-\lambda^0\|_F^2\\ \le& \frac{2}{\beta^2K^2}\|\lambda^K-\lambda^*\|_F^2+\frac{2}{\beta^2K^2}\|\lambda^0-\lambda^*\|_F^2\\ \le& \frac{4}{K^2}\left(\frac{L+\beta}{\beta}\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{\beta^2}\|\lambda^0-\lambda^*\|_F^2\right).\end{aligned} \tag{29}$$

Summing (28) over $k=1,2,\cdots,K$, dividing both sides by K, using the convexity of $\|\cdot\|_F^2$ and the definition of $\hat{\mathbf{x}}^K$, we have

$$(L+\beta)\|\hat{\mathbf{x}}^K-\mathbf{x}^*\|_F^2 \le (L+\beta)\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{2\beta}\|\lambda^0-\lambda^*\|_F^2,\quad \forall k=1,\cdots,K. \tag{30}$$

From (27), (29), (30) and Lemma 13, we have

$$\begin{aligned}F(\alpha(\hat{\mathbf{x}}^K))-F(x^*) \le& \frac{1}{mK}\left(\left(1+\frac{8L}{K\beta(1-\sigma_2(W))}\right)\left((L+\beta)\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{2\beta}\|\lambda^*\|_F^2\right)\right.\\ &+\frac{6\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}\left(\sqrt{\frac{L+\beta}{\beta}\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{\beta^2}\|\lambda^*\|_F^2}\right)\\ &\left.+\frac{4L}{\sqrt{1-\sigma_2(W)}}\sqrt{\frac{L+\beta}{\beta}}\left(\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{1}{\beta(L+\beta)}\|\lambda^*\|_F^2\right)\right).\end{aligned} \tag{31}$$

Plugging $\|\lambda^*\|_F^2 \le \frac{2\|\nabla f(\mathbf{x}^*)\|_F^2}{1-\sigma_2(W)}$, (6) and the setting of β into the above inequality, after some simple computations, we have the first conclusion. Similarly, from (29) and Lemma 10, we have the second conclusion. □

Proof of Lemma 4: For the first stage, from a modification of (24) on problem (14), we know that Algorithm 1 needs

$$K_0 = O\left(\left(\frac{L_\epsilon}{\epsilon}+\frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{1-\sigma_2(W)}\right)$$

iterations such that

$$(L_\epsilon+\beta)\|\mathbf{x}^{K_0}-\mathbf{x}_\epsilon^*\|_F^2+\frac{1}{2\beta}\|\lambda^{K_0}-\lambda_\epsilon^*\|_F^2 \le m(1-\sigma_2(W))\left(L_\epsilon R_1+R_2/L_\epsilon\right). \tag{32}$$

Let $(\mathbf{x}^{K_0},\lambda^{K_0})$ be the initialization of the second stage. From a modification of (31) on problem (12), we have

$$\begin{aligned}F_\epsilon(\alpha(\hat{\mathbf{x}}^K))-F_\epsilon(x_\epsilon^*) \le& \frac{1}{mK}\left(\left(1+\frac{8L_\epsilon}{K\beta(1-\sigma_2(W))}\right)m(1-\sigma_2(W))(L_\epsilon R_1+R_2/L_\epsilon)\right.\\ &+\frac{6\|\nabla f_\epsilon(\mathbf{x}_\epsilon^*)\|_F}{\sqrt{1-\sigma_2(W)}}\sqrt{\frac{2m(1-\sigma_2(W))(L_\epsilon R_1+R_2/L_\epsilon)}{\beta}}\\ &\left.+\frac{4L_\epsilon}{\sqrt{1-\sigma_2(W)}}\sqrt{\frac{L_\epsilon+\beta}{\beta}}\frac{2m(1-\sigma_2(W))(L_\epsilon R_1+R_2/L_\epsilon)}{L_\epsilon+\beta}\right).\end{aligned}$$

From the definition of $f_\epsilon(\mathbf{x})$, the smoothness of $f(\mathbf{x})$, Lemma 15 and (6), we have $\|\nabla f_\epsilon(\mathbf{x}_\epsilon^*)\|_F \le \|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{x}_\epsilon^* - \mathbf{x}^*\|_F + \epsilon\|\mathbf{x}_\epsilon^*\|_F \le \sqrt{mR_2} + 2L_\epsilon\sqrt{mR_1} \le \sqrt{8mL_\epsilon(L_\epsilon R_1 + R_2/L_\epsilon)}$. From $\beta = L_\epsilon$ and after some simple calculations, we have

$$F_\epsilon(\alpha(\hat{\mathbf{x}}^K)) - F_\epsilon(x_\epsilon^*) \le \frac{41(L_\epsilon R_1 + R_2/L_\epsilon)}{K}.$$

On the other hand, from Lemma 10, (29), (32) and $\beta = L_\epsilon$, we have

$$\|\Pi\hat{\mathbf{x}}^K\|_F^2 \le \frac{1}{1-\sigma_2(W)}\|U\hat{\mathbf{x}}^K\|_F^2 \le \frac{8m(R_1 + R_2/L_\epsilon^2)}{K^2}.$$

Thus, the second stage needs $K = O\left(\frac{L_\epsilon R_1 + R_2/L_\epsilon}{\epsilon}\right)$ iterations such that $F_\epsilon(\alpha(\hat{\mathbf{x}}^K)) - F_\epsilon(x_\epsilon^*) \le \epsilon$ and $\frac{1}{m}\sum_{i=1}^m \left\|\hat{x}_{(i)}^K - \alpha(\hat{\mathbf{x}}^K)\right\|^2 \le \epsilon^2$. □

4.3. Proofs of Theorems 6 and 8. We consider the strongly convex problems in Section 4.3.1 and the non-strongly convex ones in Section 4.3.2, respectively.

4.3.1. Strongly Convex Case. In this section, we prove Theorem 6. Define

$$x^{k,*} = \operatorname*{argmin}_x G^k(x) = \operatorname*{argmin}_x F(x) + \frac{\tau}{2}\|x - \alpha(\mathbf{y}^k)\|^2$$

and denote $(\mathbf{x}^{k,*}, \lambda^{k,*})$ to be a KKT point of the saddle point problem $\min_{\mathbf{x}}\max_\lambda g^k(\mathbf{x}) + \langle\lambda, U\mathbf{x}\rangle$, where $g^k(\mathbf{x}) \equiv f(\mathbf{x}) + \frac{\tau}{2}\|\mathbf{x} - \mathbf{y}^k\|_F^2$. Then, we know $\mathbf{x}^{k,*} = \mathbf{1}(x^{k,*})^T$. Let $(\mathbf{x}^{k,t}, U\lambda^{k,t})_{t=0}^{T_k+1}$ be the iterates generated by Algorithm 1 at the k-th iteration of Algorithm 2. Then, $\mathbf{x}^{k,0} = \mathbf{x}^k$ and $\mathbf{x}^{k,T_k+1} = \mathbf{x}^{k+1}$. Define

$$\rho_{k,t} = (L_g + \beta_g)\|\mathbf{x}^{k,t} - \mathbf{x}^{k,*}\|_F^2 + \frac{1}{2\beta_g}\|\lambda^{k,t} - \lambda^{k,*}\|_F^2,$$

where we set $\beta_g = L_g$. Similar to (25), we have

$$G^k(\alpha(\mathbf{x}^{k+1})) - G^k(x^{k,*}) = G^k(\alpha(\mathbf{x}^{k,T_k+1})) - G^k(x^{k,*}) \le \frac{1}{2m}\rho_{k,T_k}.$$

Thus, we only need to prove $\rho_{k,T_k} \le 2m\varepsilon_k$. Moreover, we prove a sharper result of $\rho_{k,T_k} \le 2m(1-\sigma_2(W))\varepsilon_k$ by induction in the following lemma. The reason is that we want to prove $\|\Pi\mathbf{x}^{K+1}\|_F^2 \le O(m\varepsilon_K)$ and thus we need to eliminate $1-\sigma_2(W)$ in (34).

LEMMA 17. *Suppose that Assumptions 1 and 2 hold with $\mu > 0$. If $\rho_{r,T_r} \le 2m(1-\sigma_2(W))\varepsilon_r$ holds for all $r \le k-1$ and we initialize $\mathbf{x}^{k,0} = \mathbf{x}^{k-1,T_{k-1}+1}$ and $\lambda^{k,0} = \lambda^{k-1,T_{k-1}+1}$, then we only need $T_k = O\left(\left(\frac{L_g}{\mu_g} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L_g}{\mu(1-\sigma_2(W))}\right)$ such that $\rho_{k,T_k} \le 2m(1-\sigma_2(W))\varepsilon_k$.*

Proof: From Theorem 1 and (26), we have

$$\rho_{k,T_k} \le (1-\delta_g)^{T_k}\rho_{k,0}, \tag{33}$$

$$\|\Pi\mathbf{x}^{k+1}\|_F^2 = \|\Pi\mathbf{x}^{k,T_k+1}\|_F^2 \le \frac{16}{\beta_g(1-\sigma_2(W))}\rho_{k,T_k}, \tag{34}$$

where $\delta_g = \frac{1}{39\left(\frac{L_g}{\mu_g}+\frac{1}{1-\sigma_2(W)}\right)}$. From the initialization and Theorem 1, we have

$$\begin{aligned}\rho_{k,0} =& (L_g+\beta_g)\|\mathbf{x}^{k-1,T_{k-1}+1}-\mathbf{x}^{k,*}\|_F^2+\frac{1}{2\beta_g}\|\lambda^{k-1,T_{k-1}+1}-\lambda^{k,*}\|_F^2\\ \le& 2(L_g+\beta_g)\|\mathbf{x}^{k-1,T_{k-1}+1}-\mathbf{x}^{k-1,*}\|_F^2+\frac{1}{\beta_g}\|\lambda^{k-1,T_{k-1}+1}-\lambda^{k-1,*}\|_F^2\\ &+2(L_g+\beta_g)\|\mathbf{x}^{k,*}-\mathbf{x}^{k-1,*}\|_F^2+\frac{1}{\beta_g}\|\lambda^{k,*}-\lambda^{k-1,*}\|_F^2\\ \le& 2\rho_{k-1,T_{k-1}}+2(L_g+\beta_g)\|\mathbf{x}^{k,*}-\mathbf{x}^{k-1,*}\|_F^2+\frac{1}{\beta_g}\|\lambda^{k,*}-\lambda^{k-1,*}\|_F^2.\end{aligned} \tag{35}$$

From the fact that $\mathbf{x}^{k,*}=\mathbf{1}(x^{k,*})^T$, we have

$$\begin{aligned}\|\mathbf{x}^{k,*}-\mathbf{x}^{k-1,*}\|_F^2 =& m\|x^{k,*}-x^{k-1,*}\|^2 \overset{a}{\le} m\|\alpha(\mathbf{y}^k)-\alpha(\mathbf{y}^{k-1})\|^2\\ \overset{b}{\le}& \sum_{i=1}^m\|y_{(i)}^k-y_{(i)}^{k-1}\|^2=\|\mathbf{y}^k-\mathbf{y}^{k-1}\|_F^2,\end{aligned} \tag{36}$$

where $\overset{a}{\le}$ uses Lemma 14 and $\overset{b}{\le}$ uses the definition of $\alpha(\mathbf{y})$ and the convexity of $\|\cdot\|^2$. From Lemma 11, we know

$$\|\lambda^{k,*}-\lambda^{k-1,*}\|_F^2 \le \frac{2}{1-\sigma_2(W)}\|U\lambda^{k,*}-U\lambda^{k-1,*}\|_F^2. \tag{37}$$

Recall that $(\mathbf{x}^{k,*},\lambda^{k,*})$ is a KKT point of $\min_{\mathbf{x}}\max_{\lambda} g^k(\mathbf{x})+\langle\lambda,U\mathbf{x}\rangle$ and $g^k(\mathbf{x})=f(\mathbf{x})+\frac{\tau}{2}\|\mathbf{x}-\mathbf{y}^k\|_F^2$. From the KKT condition, we have $U\lambda^{k,*}+\nabla g^k(\mathbf{x}^{k,*})=0$. Thus, we have

$$\begin{aligned}&\|U\lambda^{k,*}-U\lambda^{k-1,*}\|_F^2\\ =&\left\|\nabla f(\mathbf{x}^{k,*})+\tau(\mathbf{x}^{k,*}-\mathbf{y}^k)-\nabla f(\mathbf{x}^{k-1,*})-\tau(\mathbf{x}^{k-1,*}-\mathbf{y}^{k-1})\right\|_F^2\\ \overset{c}{\le}& 2(L+\tau)^2\|\mathbf{x}^{k,*}-\mathbf{x}^{k-1,*}\|_F^2+2\tau^2\|\mathbf{y}^k-\mathbf{y}^{k-1}\|_F^2\\ \overset{d}{\le}& 4L_g^2\|\mathbf{y}^k-\mathbf{y}^{k-1}\|_F^2,\end{aligned} \tag{38}$$

where $\overset{c}{\le}$ uses the L-smoothness of $f(\mathbf{x})$ and $\overset{d}{\le}$ uses (36) and $L_g=L+\tau$. Combing (35), (36), (37) and (38) and using $\beta_g=L_g$, we have

$$\rho_{k,0} \le 2\rho_{k-1,T_{k-1}}+\left(4L_g+\frac{8L_g}{1-\sigma_2(W)}\right)\|\mathbf{y}^k-\mathbf{y}^{k-1}\|_F^2. \tag{39}$$

From a similar induction to the proof of [18, Proposition 12] and the relations in Algorithm 2, we have

$$\begin{aligned}\|\mathbf{y}^k-\mathbf{y}^{k-1}\|_F^2 \le& 2\|\mathbf{y}^k-\mathbf{x}^*\|_F^2+2\|\mathbf{y}^{k-1}-\mathbf{x}^*\|_F^2\\ \le& 4(1+\vartheta_k)^2\|\mathbf{x}^k-\mathbf{x}^*\|_F^2+4\vartheta_k^2\|\mathbf{x}^{k-1}-\mathbf{x}^*\|_F^2\\ &+4(1+\vartheta_{k-1})^2\|\mathbf{x}^{k-1}-\mathbf{x}^*\|_F^2+4\vartheta_{k-1}^2\|\mathbf{x}^{k-2}-\mathbf{x}^*\|_F^2\\ \le& 40\max\{\|\mathbf{x}^k-\mathbf{x}^*\|_F^2,\|\mathbf{x}^{k-1}-\mathbf{x}^*\|_F^2,\|\mathbf{x}^{k-2}-\mathbf{x}^*\|_F^2\},\end{aligned} \tag{40}$$

where we denote $\vartheta_k=\frac{\theta_{k-1}(1-\theta_{k-1})}{\theta_{k-1}^2+\theta_k}$ and use $\vartheta_k\le 1,\forall k$. The later can be obtained by $\vartheta_k=\frac{\sqrt{q}-q}{\sqrt{q}+q}\le 1$ for $\mu>0$ and $\vartheta_k=\frac{\theta_{k-1}(1-\theta_{k-1})}{\theta_{k-1}^2/\theta_k}\le\frac{\theta_k}{\theta_{k-1}}\le 1$ for $\mu=0$.

Since $\rho_{r,T_r} \leq 2m\varepsilon_r$ for all $r \leq k-1$, i.e., $G^r(\alpha(\mathbf{x}^{r+1})) - G^r(x^{r,*}) \leq \varepsilon_r$, from Theorem 5 we know the following conclusion holds for all $r \leq k-1$:

$$F(\alpha(\mathbf{x}^{r+1})) - F(x^*) \leq \frac{36}{(\sqrt{q}-\rho)^2}\varepsilon_{r+1}, \tag{41}$$

where we use the definition of ε_r in Theorem 5, Thus, we have

$$\begin{aligned}\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 \overset{e}{=}&\|\mathbf{1}(\alpha(\mathbf{x}^k))^T + \Pi\mathbf{x}^k - \mathbf{1}(x^*)^T\|_F^2\\ \leq&2m\|\alpha(\mathbf{x}^k) - x^*\|^2 + 2\|\Pi\mathbf{x}^k\|_F^2\\ \overset{f}{\leq}&\frac{4m}{\mu}(F(\alpha(\mathbf{x}^k)) - F(x^*)) + \frac{32\rho_{k-1,T_{k-1}}}{\beta_g(1-\sigma_2(W))}\\ \overset{g}{\leq}&\frac{144m\varepsilon_k}{\mu(\sqrt{q}-\rho)^2} + \frac{64m\varepsilon_{k-1}}{\beta_g}\end{aligned} \tag{42}$$

where we use the definitions of $\Pi\mathbf{x}$ and $\alpha(\mathbf{x})$ in $\overset{e}{=}$, the μ-strong convexity of $F(x)$ and (34) in $\overset{f}{\leq}$, (41) and the induction condition of $\rho_{k-1,T_{k-1}} \leq 2m(1-\sigma_2(W))\varepsilon_{k-1}$ in $\overset{g}{\leq}$.

Combing (33), (39), (40), (42) and using $\rho_{k-1,T_{k-1}} \leq 2m(1-\sigma_2(W))\varepsilon_{k-1}$, we have

$$\begin{aligned}\rho_{k,T_k} \leq&(1-\delta_g)^{T_k}\varepsilon_k\left(\frac{4m}{1-\rho} + \left(4L_g + \frac{8L_g}{1-\sigma_2(W)}\right)\frac{40}{(1-\rho)^3}\left(\frac{144m\varepsilon_k}{\mu(\sqrt{q}-\rho)^2} + \frac{64m\varepsilon_{k-1}}{\beta_g}\right)\right)\\ \overset{h}{\leq}&(1-\delta_g)^{T_k}\frac{99844mL_g}{\mu(1-\sigma_2(W))(1-\rho)^3(\sqrt{q}-\rho)^2}\varepsilon_k \equiv (1-\delta_g)^{T_k}C_1\varepsilon_k,\end{aligned}$$

where we use $\sqrt{q}-\rho < 1$, $\varepsilon_k \leq \varepsilon_{k-1}$ and $\beta_g \geq \mu$ in $\overset{h}{\leq}$.

Thus, to attain $\rho_{k,T_k} \leq 2m(1-\sigma_2(W))\varepsilon_k$, we only need $(1-\delta_g)^{T_k}C_1 \leq 2m(1-\sigma_2(W))$, i.e., $T_k = O\left(\frac{1}{\delta_g}\log\frac{C_1}{2m(1-\sigma_2(W))}\right) = O\left(\left(\frac{L_g}{\mu_g} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L_g}{\mu(1-\sigma_2(W))}\right)$. □

Based on the above lemma and Theorem 6, we can prove Corollary 7.

Proof of Corollary 7: From (34) and Lemma 17, we have

$$\|\Pi\mathbf{x}^{K+1}\|_F^2 \leq \frac{32m\varepsilon_K}{\beta_g} \overset{b}{\leq} \frac{32m}{\beta_g}\frac{2(F(x^0)-F(x^*))}{9}(1-\rho)^{K+1},$$

where $\overset{b}{\leq}$ uses the definition of ε_k in Theorem 5. On the other hand, from Theorem 5, we have

$$F(\alpha(\mathbf{x}^{K+1})) - F(x^*) \leq \frac{8}{(\sqrt{q}-\rho)^2}(1-\rho)^{K+2}(F(x^0)-F(x^*)).$$

To make $\|\Pi\mathbf{x}^{K+1}\|_F^2 \leq O(m\epsilon^2)$ and $F(\alpha(\mathbf{x}^{K+1})) - F(x^*) \leq O(\epsilon)$, we only need to run Algorithm 2 for $K = O\left(\sqrt{1+\frac{\tau}{\mu}}\log\frac{1}{\epsilon}\right)$ outer iterations such that

$$(F(x^0)-F(x^*))(1-\rho)^{K+1} \leq \epsilon^2.$$

Thus, the total number of inner iterations is

$$\begin{aligned}\sum_{k=0}^{K} T_k =&\sqrt{1+\frac{\tau}{\mu}}\left(\log\frac{1}{\epsilon}\right)\left(\frac{L+\tau}{\mu+\tau} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{L+\tau}{\mu(1-\sigma_2(W))}\\ \leq&3\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{2L}{\mu(1-\sigma_2(W))}\log\frac{1}{\epsilon}\end{aligned}$$

by letting $\tau = L(1-\sigma_2(W)) - \mu$.

4.3.2. Non-strongly Convex Case. When the strong-convexity is absent, we can have the following lemma, which further leads to Theorem 8. Similar to Lemma 17, we prove a sharper result of $\rho_{k,T_k} \leq 2m\varepsilon_k(1-\sigma_2(W))^{3+\xi}$.

LEMMA 18. *Suppose that $F(x)$ is convex. If $\rho_{r,T_r} \leq 2m\varepsilon_r(1-\sigma_2(W))^{3+\xi}$ holds for all $r \leq k-1$ and we initialize $\mathbf{x}^{k,0} = \mathbf{x}^{k-1,T_{k-1}+1}$ and $\lambda^{k,0} = \lambda^{k-1,T_{k-1}+1}$, then we only need $T_k = O\left(\left(\frac{L_g}{\mu_g} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{k}{1-\sigma_2(W)}\right)$ such that $\rho_{k,T_k} \leq 2m\varepsilon_k(1-\sigma_2(W))^{3+\xi}$.*

The proof is similar to that of [18, Proposition 12] and we omit the details. Simply, when the strong convexity is absent, (39) and (40) also hold. But we need to bound $\|\mathbf{x}^k - \mathbf{x}^*\|_F^2$ in a different way. From Theorem 5, the sequence $F(\alpha(\mathbf{x}^k))$ is bounded by a constant. By the bounded level set assumption, there exists $C > 0$ such that $\|\alpha(\mathbf{x}^k) - x^*\| \leq C$. From (34), we have $\|\Pi\mathbf{x}^k\|_F^2 \leq \frac{16}{\beta_g(1-\sigma_2(W))}\rho_{k-1,T_{k-1}}$. Thus, we have

$$\begin{aligned}\|\mathbf{x}^k - \mathbf{x}^*\|_F^2 =& \|\mathbf{1}(\alpha(\mathbf{x}^k))^T + \Pi\mathbf{x}^k - \mathbf{1}(x^*)^T\|_F^2\\ \leq& 2m\|\alpha(\mathbf{x}^k) - x^*\|^2 + 2\|\Pi\mathbf{x}^k\|_F^2\\ \leq& 2mC^2 + \frac{32m\varepsilon_{k-1}(1-\sigma_2(W))^{2+\xi}}{\beta_g}.\end{aligned}$$

Thus, $\rho_{k,0}$ is bounded by constant $C_2 = 4m + 40\left(4L_g + \frac{8L_g}{1-\sigma_2(W)}\right)(2mC^2 + 32m/\beta_g)$ and we only need $(1-\delta_g)^{T_k}C_2 \leq 2m\varepsilon_k(1-\sigma_2(W))^{3+\xi}$, i.e., $T_k = O\left(\frac{1}{\delta_g}\log\frac{C_2}{2m\varepsilon_k(1-\sigma_2(W))^{3+\xi}}\right)$ $= O\left(\left(\frac{L_g}{\mu_g} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{k}{1-\sigma_2(W)}\right)$.

Now, we come to Corollary 9. From Theorem 5, to find an ϵ-optimal solution such that $F(\alpha(\mathbf{x}^{K+1})) - F(x^*) \leq \epsilon$, we need $K = O\left(\sqrt{\frac{\tau R_1}{\epsilon}}\right)$ outer iterations. On the other hand, from (34) and Lemma 18, we have

$$\|\Pi\mathbf{x}^{K+1}\|_F^2 \leq \frac{32m(1-\sigma_2(W))^{2+\xi}\varepsilon_K}{\beta_g} \overset{a}{\leq} \frac{32m(1-\sigma_2(W))^{2+\xi}}{\beta_g K^{4+2\xi}} \overset{b}{\leq} \frac{32m\epsilon^2}{\beta_g L^{2+\xi}R_1^{2+\xi}},$$

where $\overset{a}{\leq}$ uses the definition ε_k in Theorem 5, $\overset{b}{\leq}$ uses $K = O\left(\sqrt{\frac{\tau R_1}{\epsilon}}\right)$ and $\tau = L(1-\sigma_2(W))$. Thus, the settings of T_k and K leads to $\|\Pi\mathbf{x}^{K+1}\|_F^2 \leq O(m\epsilon^2)$. The total number of inner iterations is

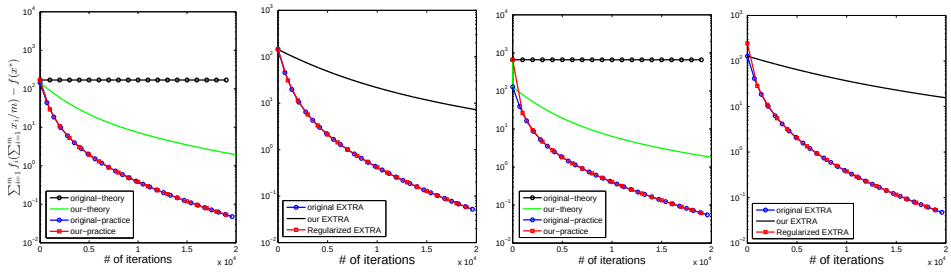
$$\sum_{k=0}^{\sqrt{\frac{\tau}{\epsilon}}} T_k = \sqrt{\frac{\tau}{\epsilon}}\left(\frac{L+\tau}{\tau} + \frac{1}{1-\sigma_2(W)}\right)\log\frac{1}{\epsilon(1-\sigma_2(W))}.$$

The setting of $\tau = L(1-\sigma_2(W))$ leads to the minimal value of $\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon(1-\sigma_2(W))}$.

5. Numerical Experiments. Consider the decentralized least square problem:

$$\min_{x\in\mathbb{R}^n}\sum_{i=1}^m f_i(x) \quad \text{with} \quad f_i(x) \equiv \frac{1}{2}\|A_i^T x - b_i\|^2 + \frac{\mu}{2}\|x\|^2, \tag{43}$$

where each agent $\{1,\cdots,m\}$ holds its own local function $f_i(x)$. $A_i \in \mathbb{R}^{n\times s}$ is generated from the uniform distribution with each entry in $[0,1]$ and each column of A_i is normalized to be 1. We set $s = 10$, $n = 500$, $m = 100$ and $b_i = A_i^T x$ with some unknown x. We test the performance of the proposed algorithms on both strongly convex problem and non-strongly



(a). SC, Erdős-Rényi graph (b). NS, Erdős-Rényi graph (c). SC, Geometric graph (d). NS, Geometric graph

FIG. 1. *Comparisons between different EXTRA on the Erdős-Rényi random graph (p=0.1) and the geometric graph (d=0.3). SC means the strongly convex problem ($\mu = 10^{-6}$) and NS means the non-strongly convex one.*

convex one. For the strongly convex case, we test on $\mu = 10^{-6}$ and $\mu = 10^{-8}$, respectively. In general, the accelerated algorithms apply to ill-conditioned problems with large condition numbers. For the non-strongly convex one, we let $\mu = 0$.

We test the performance on two kinds of network: (1), the Erdős-Rényi random graph where each pair of nodes has a connection with the ratio of p. We test two different settings of p: $p = 0.5$ and $p = 0.1$, which results to $\frac{1}{1-\sigma_2(W)} = 2.87$ and $\frac{1}{1-\sigma_2(W)} = 7.74$, respectively. (2), the geometric graph where m nodes are placed uniformly and independently in the unit square $[0, 1]$ and two nodes are connected if their distance is at most d. We test on $d = 0.5$ and $d = 0.3$, which leads to $\frac{1}{1-\sigma_2(W)} = 8.13$ and $\frac{1}{1-\sigma_2(W)} = 30.02$, respectively. We set the weight matrix as $W = \frac{I+\mathbf{M}}{2}$ for both graphes, where $\mathbf{M}$ is the Metropolis weight matrix [6]: $\mathbf{M}_{i,j}=\begin{cases}1/(1+\max\{d_i,d_j\}), & \text{if } (i,j)\in\mathcal{E},\\ 0, & \text{if } (i,j)\notin\mathcal{E} \text{ and } i\neq j, \text{ and } d_i \text{ is the number of the } i\text{-th}\\ 1-\sum_{l\in\mathcal{N}_i} W_{i,l}, & \text{if } i=j,\end{cases}$ agent's neighbors.

We first compare EXTRA analyzed in this paper with the original EXTRA [34]. For the strongly convex problem, [34, Remark 4] analyzed the algorithm with $\alpha = \frac{1}{\beta} = \frac{\mu^2}{L}$ and $\alpha = \frac{1}{\beta} = \frac{1}{L}$ is suggested in practice. In our theory, we use $\beta = L$ and $\alpha = \frac{1}{4L}$. In practice, we observe that $\beta = L$ and $\alpha = \frac{1}{L}$ performs the best. Figure 1.a and Figure 1.c plot the results. We can see that the theoretical setting in the original EXTRA makes almost no decreasing in the objective function values due to small step-size and our theoretical setting works much better. On the other hand, both the original EXTRA and our analyzed one work best for $\beta = L$ and $\alpha = \frac{1}{L}$. For the non-strongly convex problems, [34] suggests $\alpha = \frac{1}{\beta} = \frac{1}{L}$ in both theory and practice. In our theory, Lemma 3 suggests $\beta = \frac{L}{\sqrt{1-\sigma_2(W)}}$ and $\alpha = \frac{1}{2(L+\beta)}$. From Figure 1.b and Figure 1.d, we observe that a larger β (i.e., a smaller step-size) makes the algorithm slow. On the other hand, our regularized EXTRA performs as well as the original EXTRA.

Then, we compare the proposed accelerated EXTRA (Acc-EXTRA) with the original EXTRA [34], the accelerated distributed Nesterov gradient descent (Acc-DNGD) [27], accelerated dual ascent (ADA) [38] and the accelerated penalty method with consensus (APM-C) [17]. For the strongly convex problem, we set $\tau = L(1-\sigma_2(W)) - \mu$ and $T_k = \left\lceil \frac{1}{5(1-\sigma_2(W))} \log \frac{L}{\mu(1-\sigma_2(W))} \right\rceil$ for Acc-EXTRA, $T_k = \left\lceil \frac{k\sqrt{\mu/L}}{4\sqrt{1-\sigma_2(W)}} \right\rceil$ and the step-size as $\frac{1}{L}$ for APM-C, $T_k = \left\lceil \sqrt{\frac{L}{\mu}} \log \frac{L}{\mu} \right\rceil$ and the step-size as μ for ADA, where T_k means the number of inner iterations at the k-th outer iteration and $\lceil\cdot\rceil$ is the top integral function. We

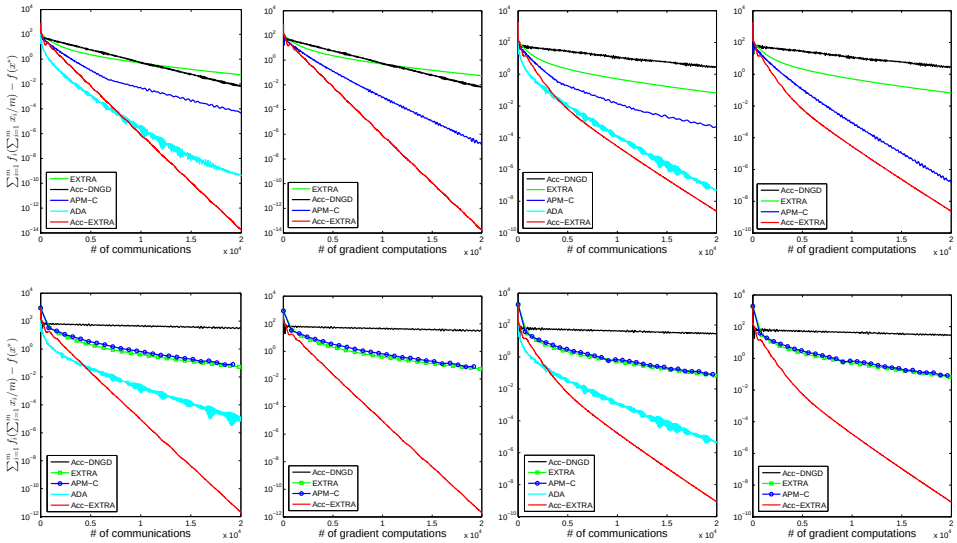


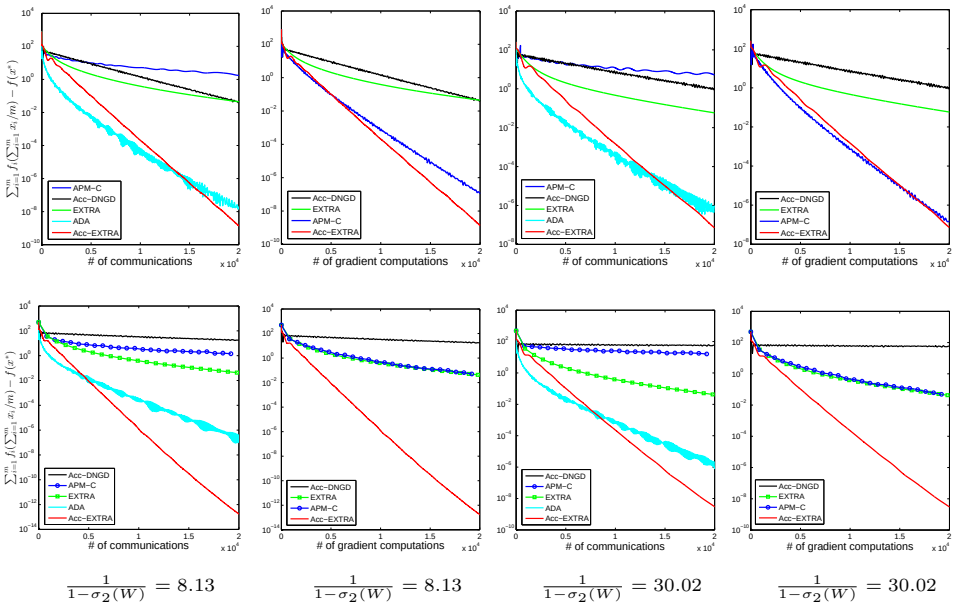
FIG. 2. *Comparisons on the strongly convex problem with the Erdős-Rényi random graph.* $p = 0.5$ *for the left two plots and* $p = 0.1$ *for the right two.* $\mu = 10^{-6}$ *for the top four plots and* $\mu = 10^{-8}$ *for the bottom four.*

set the step-size as $\frac{1}{L}$ for EXTRA and tune the best step-size for Acc-DNGD with different graphs and different μ. All the compared algorithms start from $x_{(i)} = \mathbf{0}, \forall i$.

The numerical results are illustrated in Figures 2 and 3. The computation cost of ADA is high and it has almost no visible decreasing in the first $20,000$ gradient computations [17, Figure 2]. Thus, we do not paint it in the second and forth plots of Figures 2-5. We can see that Acc-EXTRA performs better than the original EXTRA on both the Erdős-Rényi random graph and the geometric graph. We also observe that Acc-EXTRA is superior to ADA and APM-C on the graphs with small μ and $\frac{1}{1-\sigma_2(W)}$. The performance of Acc-EXTRA degenerates when μ and $\frac{1}{1-\sigma_2(W)}$ become larger. When preparing the experiments, we observe that Acc-EXTRA applies to ill-conditioned problems with large condition numbers for strongly convex problems. In this case, Acc-EXTRA runs with a certain number of outer iterations and the acceleration takes effect.

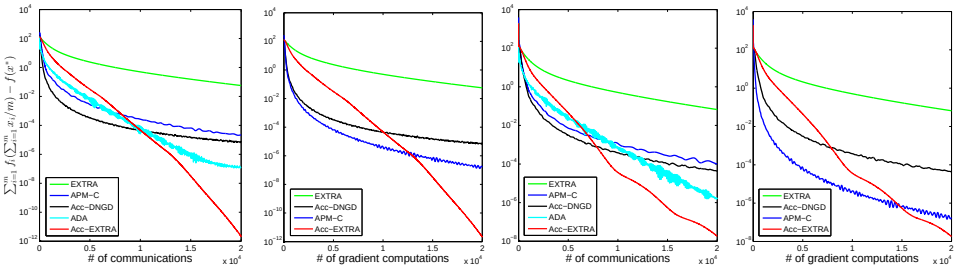
For the non-strongly convex problem ($\mu = 0$), we set $\tau = L(1 - \sigma_2(W))$ and $T_k = \left\lceil \frac{1}{2(1-\sigma_2(W))} \log \frac{k+1}{1-\sigma_2(W)} \right\rceil$ for Acc-EXTRA, $T_k = \left\lceil \frac{\log(k+1)}{5\sqrt{1-\sigma_2(W)}} \right\rceil$ and the step-size as $\frac{1}{L}$ for APM-C. We tune the best step-size as $\frac{1}{L}$ and $\frac{0.2}{L}$ for EXTRA and Acc-DNGD, respectively. For ADA, we add a small regularizer of $\frac{\epsilon}{2}\|x\|^2$ to each $f_i(x)$ and solve a regularized strongly covnex problem with $\epsilon = 10^{-7}$. The numerical results are illustrated in Figures 4 and 5. We observe that Acc-EXTRA also outperforms the original EXTRA and Acc-EXTRA is superior with small $\frac{1}{1-\sigma_2(W)}$. Moreover, at the first 10000 iterations, the advantage of Acc-EXTRA is not obvious and it performs better at the last 5000 iterations. Thus, Acc-EXTRA suits for the applications requiring a high precision and the well-connected networks with small $\frac{1}{1-\sigma_2(W)}$.

At last, we report two results in Figure 6 that Acc-EXTRA does not perform well, where the left two plots are for the strongly convex problem and the right two are for the non-strongly convex one. Comparing the left two plots in Figure 6 with the left and top two in Figure 3, we can see that Acc-EXTRA is inferior to ADA and APM-C in cases with a larger μ, i.e., a



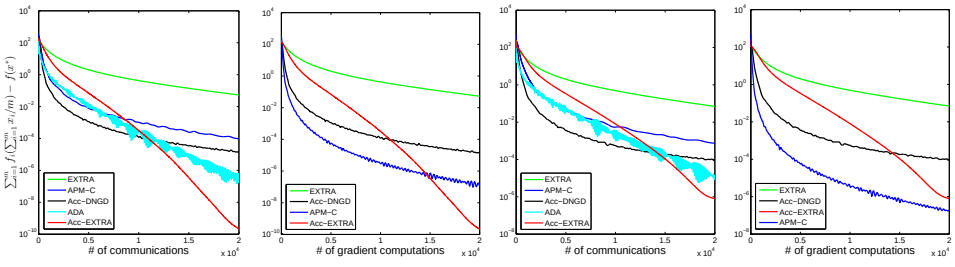
$\frac{1}{1-\sigma_2(W)} = 8.13$ $\frac{1}{1-\sigma_2(W)} = 8.13$ $\frac{1}{1-\sigma_2(W)} = 30.02$ $\frac{1}{1-\sigma_2(W)} = 30.02$

FIG. 3. *Comparisons on the strongly convex problem with the geometric graph.* $d = 0.5$ *for the left two plots and* $d = 0.3$ *for the right two.* $\mu = 10^{-6}$ *for the top four plots and* $\mu = 10^{-8}$ *for the bottom four.*



$\frac{1}{1-\sigma_2(W)} = 2.87$ $\frac{1}{1-\sigma_2(W)} = 2.87$ $\frac{1}{1-\sigma_2(W)} = 7.74$ $\frac{1}{1-\sigma_2(W)} = 7.74$

FIG. 4. *Comparisons on the non-strongly convex problem with the Erdős-Rényi random graph.* $p = 0.5$ *for the left two plots and* $p = 0.1$ *for the right two plots.*



$\frac{1}{1-\sigma_2(W)} = 8.13$ $\frac{1}{1-\sigma_2(W)} = 8.13$ $\frac{1}{1-\sigma_2(W)} = 30.02$ $\frac{1}{1-\sigma_2(W)} = 30.02$

FIG. 5. *Comparisons on the non-strongly convex problem with the geometric graph.* $d = 0.5$ *for the left two plots and* $d = 0.3$ *for the right two plots.*

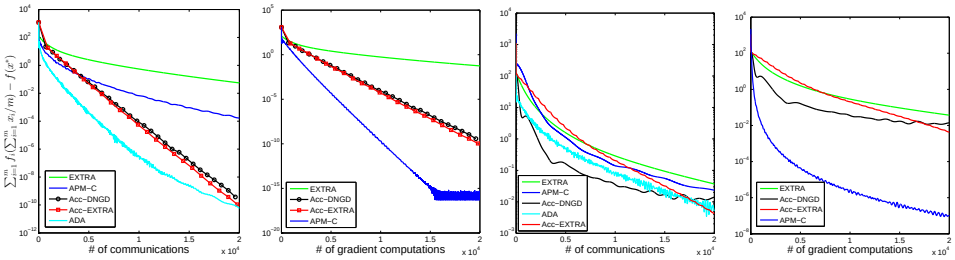


FIG. 6. *Further comparisons on the geometric graph.* $d = 0.5$ *and* $\mu = 10^{-5}$ *for the left two plots and* $d = 0.15$ *and* $\mu = 0$ *for the right two plots.*

smaller condition number for strongly convex problems. On the other hand, comparing the right two in Figure 6 with the four in Figure 5, we observe that ADA and APM-C outperforms Acc-EXTRA in cases with a larger $\frac{1}{1-\sigma_2(W)}$ (it equals 268.67 when $d = 0.15$) for non-strongly convex problems. These observations further support the above conclusions.

6. Conclusion. In this paper, we first give a sharp analysis on the original EXTRA with improved complexities, which depends on the sum of $\frac{L}{\mu}$ (or $\frac{L}{\epsilon}$) and $\frac{1}{1-\sigma_2(W)}$, rather than their product. Then, we use the Catalyst framework to accelerate it and obtain the near optimal communication complexities and competitive computation complexities. Our communication complexities of the proposed accelerated EXTRA are only worse by the factors of $\left(\log \frac{L}{\mu(1-\sigma_2(W))}\right)$ and $\left(\log \frac{1}{\epsilon}\right)$ form the lower bounds for strongly convex and non-strongly convex problems, respectively.

REFERENCES

[1] B. Recht abd C. Re, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradietn descent. In *Advances in Neural Information Processing Systems (NIPS)*, pages 693–701, 2011.

[2] A. Agarwal and J. Duchi. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 873–881, 2011.

[3] Z. Allen-Zhu and E. Hazan. Optimal black-box reductions between optimization objectives. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1614–1622, 2016.

[4] N. Aybat, Z. Wang, T. Lin, and S. Ma. Distributed linearized alternating direction method of multipliers for composite convex consensus optimization. *IEEE Transaction on Automatic Control*, 63(1):50–20, 2018.

[5] D. Bertsekas. Distributed asynchromous computation of fixed points. *Mathmatical Programming*, 27:107–120, 1983.

[6] S. Boyd, P. Diaconis, and L. Xiao. Fastest mixing markov chain on a graph. *SIAM Review*, 46(4):667–689, 2004.

[7] A. Chen and A. Ozdaglar. A fast distributed proximal-gradient method. In *Allerton Conference on Communication, Control, and Computing*, pages 601–608, 2012.

[8] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(6):165–202, 2012.

[9] M. Duarte and Y. Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2014.

[10] P. Forero, A. Cano, and G. Giannakis. Consensus-based distributed support vector machines. *Journal of Machine Learning Research*, 59(55):1663–1707, 2010.

[11] L. Gan, U. Topcu, and S. Low. Optimal decentralized protocol for electric vehicle charging. *IEEE Transaction on Power Systems*, 28(2):940–951, 2013.

[12] M. Hong, D. Hajinezhad, and M. Zhao. Prox-PDA: The proximal primal-dual algorithm for fast distributed nonconvex optimization and learning over networks. In *International Conference on Machine Learning (ICML)*, pages 1529–1538, 2017.

[13] F. Iutzeler, P. Bianchi, P. Ciblat, and W. Hachem. Explicit convergence rate of a distributed alternating direction method of multipliers. *IEEE Transaction on Automatic Control*, 61(4):892–904, 2016.

[14] D. Jakovetić. A unification and generatliztion of exact distributed first order methods. *IEEE Transactions on Signal and Information Processing over Networks*, 5(1):31–46, 2019.
[15] D. Jakovetic, J. Xavier, and J. Moura. Fast distributed gradient methods. *IEEE Transaction on Automatic Control*, 59(5):1131–1146, 2014.
[16] G. Lan, S. Lee, and Y. Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. *Mathematical Programming*, 180(1):237–284, 2020.
[17] H. Li, C. Fang, W. Yin, and Z. Lin. A sharp convergence rate analysis for distributed accelerated gradient methods. *arxiv:1810.01053*, 2018.
[18] H. Lin, J. Mairal, and Z. Harchaoui. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18(212):1–54, 2018.
[19] A. Makhdoumi and A. Ozdaglar. Convergence rate of distributed ADMM over networks. *IEEE Transaction on Automatic Control*, 62(10):5082–5095, 2017.
[20] A. Mokhtari and A. Ribeiro. DSA: Decenrtalized double stochastic averaging gradient algorithm. *Journal of Machine Learning Research*, 17(61):1–35, 2016.
[21] A. Nedić. Asynchronous broadcast-based convex optimization over a network. *IEEE Transaction on Automatic Control*, 56(6):1337–1351, 2011.
[22] A. Nedić, A. Olshevsky, and M. Rabbat. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106(5):953–976, 2018.
[23] A. Nedić, A. Olshevsky, and W. Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal of Optimization*, 27(4):2597–2633, 2017.
[24] A. Nedić and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transaction on Automatic Control*, 54(1):48–61, 2009.
[25] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.
[26] Y. Nesterov. Introductory lectures on convex optimization. a basic course. *Kluwer, Boston*, 2014.
[27] G. Qu and N. Li. Accelerated distributed nesterov gradient descent. *arxiv:1705.07176*, 2017.
[28] G. Qu and N. Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2018.
[29] S. Ram, A. Nedić, and V. Veeravalli. Distributed stochastic subgradient projection algorithms for convex optimization. *Journal of Optimization Theory and Applications*, 147(3):516–545, 2010.
[30] K. Scaman, F. Bach, S. Bubeck, Y. Lee, and L. Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *International Conference on Machine Learning (ICML)*, pages 3027–3036, 2017.
[31] K. Scaman, F. Bach, S. Bubeck, Y. Lee, and L. Massoulié. Optimal algorithms for non-smooth distributed optimization in networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2740–2749, 2018.
[32] K. Scaman, F. Bach, S. Bubeck, Y. Lee, and L. Massoulié. Optimal convergence rates for convex distributed optimization in networks. *Journal of Machine Learning Research*, 20(159):1–31, 2019.
[33] M. Schmidt, N. Le Roux, and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1458–1466, 2011.
[34] W. Shi, Q. Ling, G. Wu, and W. Yin. EXREA: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.
[35] W. Shi, Q. Ling, G. Wu, and W. Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Transaction on Signal Processing*, 63(23):6013–6023, 2015.
[36] H. Terelius, U. Topcu, and R. Murray. Decentralized multi-agent optimization via dual decomposition. *IFAC proceedings volumes*, 44(1):11245–11251, 2011.
[37] J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transaction on Automatic Control*, 31(9):803–812, 1986.
[38] C. Uribe, S. Lee, A. Gasnikov, and A. Nedić. A dual approach for optimal algorithms in distributed optimization over networks. *arxiv:1809.00710*, 2018.
[39] J. Xu, S. Zhu, Y. Soh, and L. Xie. Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes. In *IEEE Conference on Decision and Control (CDC)*, pages 2055–2060, 2015.
[40] K. Yuan, Q. Ling, and W. Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.