Decentralized Accelerated Gradient Methods With Increasing Penalty Parameters

Huan Li, *Member, IEEE,* Cong Fang, Wotao Yin, Zhouchen Lin, *Fellow, IEEE,*

***Abstract*—In this paper, we study the communication and (sub)gradient computation costs in distributed optimization. We present two algorithms based on the framework of the accelerated penalty method with increasing penalty parameters. Our first algorithm is for smooth distributed optimization and it obtains the near optimal $O(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon})$ communication complexity and the optimal $O(\sqrt{\frac{L}{\epsilon}})$ gradient computation complexity for L-smooth convex problems, where $\sigma_2(W)$ denotes the second largest singular value of the weight matrix W associated to the network and ϵ is the target accuracy. When the problem is μ-strongly convex and L-smooth, our algorithm has the near optimal $O(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log^2\frac{1}{\epsilon})$ complexity for communications and the optimal $O(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon})$ complexity for gradient computations. Our communication complexities are only worse by a factor of $(\log\frac{1}{\epsilon})$ than the lower bounds. Our second algorithm is designed for nonsmooth distributed optimization and it achieves both the optimal $O(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}})$ communication complexity and $O(\frac{1}{\epsilon^2})$ subgradient computation complexity, which match the lower bounds for nonsmooth distributed optimization.**

***Index Terms*—Distributed accelerated gradient algorithms, accelerated penalty method, optimal (sub)gradient computation complexity, near optimal communication complexity.**

I. Introduction

In this paper, we consider the following distributed convex optimization problem:

$$\min_{x\in\mathbb{R}^n}\frac{1}{m}\sum_{i=1}^{m}F_i(x)\equiv f_i(x)+h_i(x), \tag{1}$$

where m agents form a connected and undirected network $\mathcal{G}=(\mathcal{V},\mathcal{E})$, $\mathcal{V}=\{1,2,...,m\}$ is the set of agents and $\mathcal{E}\subset\mathcal{V}\times\mathcal{V}$ is the set of edges, F_i is the local objective function only available to agent i and x is the decision variable. f_i is a convex and smooth function while h_i is a convex but possibly nonsmooth one. We consider distributed algorithms using only local computations and communications, i.e., each agent i makes its decision only based on the local computations on F_i (i.e., the gradient of f_i and the subgradient of h_i) and the local information received from its neighbors in the network. A pair of agents can exchange information if and only if they are directly connected in the network. Distributed computation has been widely used in signal processing [1], automatic control [2], [3] and machine learning [4]–[6].

A. Literature Review

Among the classical distributed first-order algorithms, two different types of methods have been proposed, namely, the primal-only methods and the dual-based methods.

The distributed subgradient method is a representative primal-only distributed optimization algorithm over general networks [14], while its stochastic version was studied in [15], and asynchronous variant in [16]. In the distributed subgradient method, each agent performs a consensus step and then follows a subgradient descent with a diminishing step-size. To avoid the diminishing step-size, three different types of methods have been proposed. The first type of methods [7], [17]–[19] rely on tracking differences of gradients, which keep a variable to estimate the average gradient and use this estimation in the gradient descent step. The second type of methods, called EXTRA [20], [21], introduce two different weight matrices as opposed to a single one with the standard distributed gradient method [14]. EXTRA also uses the gradient tracking. The third type of methods employ a multi-consensus inner loop [8], [22] and thus improve the consensus of the variables at each outer iteration.

The dual-based methods introduce the Lagrangian function and work in the dual space. Many classical methods can be used to solve the dual problem, e.g., the dual subgradient ascent [23], dual gradient ascent [24], accelerated dual gradient ascent [9], [12], the primal-dual method [13], [25], and ADMM [26]–[31]. In general, most dual-based methods require the evaluation of the Fenchel conjugate of the local objective function $f_i(x)$ and thus have a larger gradient computation cost per iteration than the primal-only algorithms for smooth distributed optimization. For nonsmooth problems, the authors of [11], [13], [25] studied the communication-efficient primal-dual method. Specifically, they use the classical primal-dual method [32] in the outer loop and the subgradient method in the inner loop. The authors of [13] used Chebyshev acceleration [33] to further reduce the

H. Li is sponsored by Zhejiang Lab (grant no. 2019KB0AB02). Z. 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.

H. Li is with the Institute of Robotics and Automatic Information Systems, College of Artificial Intelligence, Nankai University, Tianjin, China (lihuan_ss@126.com), and the the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China. This work was done when Li was a Ph.D student at the Key Lab. of Machine Perception (MOE), School of EECS, Peking University, Beijing, China.

C. Fang is with the Department of Electrical Engineering, Princeton University, Princeton, New Jersey, USA (fangcong@pku.edu.cn).

W. Yin is with the Department of Mathematics, University of California, Los Angeles, USA (wotaoyin@math.ucla.edu).

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

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes some additional experimental results. This material is 2M in size.

Non-strongly convex and smooth case		
Methods	Complexity of gradient computations	Complexity of communications
DNGD[1]	$O\left(\frac{1}{\epsilon^{5/7}}\right)$ [7]	$O\left(\frac{1}{\epsilon^{5/7}}\right)$ [7]
DN-C	$O\left(\sqrt{\frac{L}{\epsilon}}\right)$ [8]	$O\left(\sqrt{\frac{L}{\epsilon}}\frac{1}{1-\sigma_2(W)}\log\frac{1}{\epsilon}\right)$ [8]
Accelerated Dual Ascent	$O\left(\frac{L}{\epsilon\sqrt{1-\sigma_2(W)}}\log^2\frac{1}{\epsilon}\right)$ [9]	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [9]
Our APM-C	$O\left(\sqrt{\frac{L}{\epsilon}}\right)$	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$
Lower Bound	$O\left(\sqrt{\frac{L}{\epsilon}}\right)$ [10]	$O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\right)$ [11]
Strongly convex and smooth case		
Methods	Complexity of gradient computations	Complexity of communications
DNGD	$O\left(\left(\frac{L}{\mu}\right)^{5/7}\frac{1}{(1-\sigma_2(W))^{1.5}}\log\frac{1}{\epsilon}\right)$ [7]	$O\left(\left(\frac{L}{\mu}\right)^{5/7}\frac{1}{(1-\sigma_2(W))^{1.5}}\log\frac{1}{\epsilon}\right)$ [7]
Accelerated Dual Ascent	$O\left(\frac{L}{\mu\sqrt{1-\sigma_2(W)}}\log^2\frac{1}{\epsilon}\right)$ [3] [9]	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [9], [12]
Our APM-C	$O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log^2\frac{1}{\epsilon}\right)$
Lower Bound	$O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ [10]	$O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ [12]
Convex and Nonsmooth case		
Methods	Complexity of subgradient computations	Complexity of communications
Primal-dual method	$O\left(\frac{1}{\epsilon^2}\right)$ [11]	$O\left(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$ [11]
Smoothed accelerated gradient sliding method	$O\left(\frac{1}{\epsilon^2}\right)$ [9]	$O\left(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$ [9]
Our APM	$O\left(\frac{1}{\epsilon^2}\right)$	$O\left(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$
Lower Bound	$O\left(\frac{1}{\epsilon^2}\right)$ [13]	$O\left(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$ [13]

TABLE I
COMPLEXITY COMPARISONS BETWEEN ACCELERATED DUAL ASCENT, DN-C, DNGD, THE PRIMAL-DUAL METHOD AND OUR METHODS (APM-C,APM) FOR DISTRIBUTED CONVEX PROBLEMS.

computation complexity while the authors of [11] did it via carefully setting the parameters.

Among the methods described above, the distributed Nesterov gradient with consensus iterations (D-NC) proposed in [8] and the distributed Nesterov gradient descent (DNGD) proposed in [7] employ Nesterov's acceleration technique in the primal space, and the accelerated dual ascent proposed in [12] use the standard accelerated gradient descent in the dual space. Moreover, D-NC attains the optimal gradient computation complexity for nonstrongly convex and smooth problems, and the accelerated dual ascent achieves the optimal communication complexity for strongly convex and smooth problems, which match the complexity lower bounds [10], [12]. For nonsmooth problems, the primal-dual method proposed in [11], [13] and the smoothed accelerated gradient sliding method in [9] achieve both the optimal communication and subgradient computation complexities, which also match the lower bounds [13]. We denote the communication and computation complexities as the numbers of communications and (sub)gradient computations to find an ϵ-optimal solution x such that $\frac{1}{m}\sum_{i=1}^m F_i(x) - \min_x \frac{1}{m}\sum_{i=1}^m F_i(x) \leq \epsilon$, respectively.

B. Contributions

In this paper, we study the decentralized accelerated gradient methods with near optimal complexities from the perspective of the accelerated penalty method. Specifically, we propose an *A*ccelerated *P*enalty *M*ethod with increasing penalties for smooth distributed optimization by employing a multi-*C*onsensus inner loop (APM-C). The theoretical significance of our method is that we show the near optimal communication complexities and the optimal gradient computation complexities for both strongly convex and nonstrongly convex problems. Our communication complexities are only worse by a logarithm factor than the lower bounds.

Table I summarizes the complexity comparisons to the state-of-the-art distributed optimization algorithms (the notations in Table I will be specified precisely soon), namely, DNGD, D-NC, and the accelerated dual ascent reviewed above, as well as the complexity lower bounds. Our complexities match the lower bounds except that the communication ones have an extra factor of $\log\frac{1}{\epsilon}$. The communication complexity of the accelerated dual ascent matches ours for nonstrongly convex problems and is optimal for strongly convex problems (thus better than ours by $\log\frac{1}{\epsilon}$). On the other hand, our gradient computation complexities match the lower bounds and they are better than the compared methods. It should be noted that due to term $\log^2\frac{1}{\epsilon}$, our communication complexity for strongly convex problems is not a linear convergence rate.

Our framework of accelerated penalty method with increasing penalties also applies to nonsmooth distributed optimization. It drops the multi-consensus inner loop but employs an inner loop with several runs of subgradient method. Both the optimal communication and subgradient computation complexities are achieved, which match the lower bounds for nonsmooth distributed optimization. Although the theoretical complexities are the same with the methods [9], [11], our method gives the users a new choice in practice.

[1]The authors of [7] did not give the dependence on $1-\sigma_2(W)$. It does not mean that their complexity has no dependence on $1-\sigma_2(W)$.

C. Notations and Assumptions

Throughout the paper, the variable $x \in \mathbb{R}^n$ is the decision variable of the original problem (1). We denote $x_{(i)} \in \mathbb{R}^n$ to be the local estimate of the variable x for agent i. To simplify the algorithm description in a compact form, we introduce the aggregate variable $\mathbf{x}$, aggregate objective function $f(\mathbf{x})$ and aggregate gradient $\nabla f(\mathbf{x})$ as

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

where $\mathbf{x} \in \mathbb{R}^{m\times n}$, whose value at iteration k is denoted by $\mathbf{x}^k$. For the double loop algorithms, we denote $\mathbf{x}^{k,t}$ as its value at the kth outer iteration and tth inner iteration. Assume that the set of minimizers is non-empty. Denote x^* as one minimizer of problem (1), and let $\mathbf{x}^* = \mathbf{1}(x^*)^T \in \mathbb{R}^{m\times n}$, where $\mathbf{1} = (1, 1, \cdots, 1)^T \in \mathbb{R}^m$ is the vector with all ones. Denote $\partial h_i(x)$ as the subdifferential of $h_i(x)$ at x, and specifically, $\hat{\nabla} h_i(x) \in \partial h_i(x)$ as its one subgradient. For h_i, we introduce its aggregate objective function $h(\mathbf{x})$ and aggregate subgradient $\hat{\nabla} h(\mathbf{x})$ as

$$h(\mathbf{x}) = \sum_{i=1}^m h_i(x_{(i)}) \quad \text{and} \quad \hat{\nabla} h(\mathbf{x}) = \begin{pmatrix} \hat{\nabla} h_1(x_{(1)})^T \\ \vdots \\ \hat{\nabla} h_m(x_{(m)})^T \end{pmatrix}.$$

We use $\|\cdot\|$ and $\|\cdot\|_1$ as the l_2 Euclidean norm and l_1 norm for a vector, respectively. For matrices $\mathbf{x}$ and $\mathbf{y}$, we denote $\|\mathbf{x}\|_F$ as the Frobenius norm, $\|\mathbf{x}\|_2$ as the spectral norm and $\langle \mathbf{x}, \mathbf{y}\rangle = \text{trace}(\mathbf{x}^T\mathbf{y})$ as their inner product. Denote $I \in \mathbb{R}^{m\times m}$ as the identity matrix and $\mathcal{N}_i$ as the neighborhood of agent i in the network. Define

$$\alpha(\mathbf{x}) = \tfrac{1}{m}\textstyle\sum_{i=1}^m x_{(i)} \tag{2}$$

as the average across the rows of $\mathbf{x}$. Define two operators

$$\Pi = I - \tfrac{1}{m}\mathbf{1}\mathbf{1}^T \quad \text{and} \quad U = \sqrt{I - W} \tag{3}$$

to measure the consensus violation, where W is the weight matrix associated to the network, which describes the information exchange through the network. Especially, $\|\Pi\mathbf{x}\|_F$ directly measures the distance between $x_{(i)}$ and $\alpha(\mathbf{x})$. We follow [12] to define $\sqrt{A} = V\sqrt{\Lambda}V^T$, given the eigenvalue decomposition $A = V\Lambda V^T$ of the symmetric positive semidefinite matrix A.

We make the following assumptions for each function $f_i(x)$.

Assumption 1:

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

In Assumption 1, μ and L are the strong-convexity constant and smoothness constant, respectively. Assumption 1 yields that the aggregate function $f(\mathbf{x})$ is also μ-strongly convex and L-smooth. For the nonsmooth function $h_i(x)$, we follow [25] to make the following assumptions.

Assumption 2:

1) $h_i(x)$ is convex.
2) $h_i(x)$ is M-Lipschitz continuous: $h_i(y) \leq h_i(x) + \left\langle \hat{\nabla} h_i(x), y - x \right\rangle + M\|y - x\|$.

We can simply verify that $h(\mathbf{x})$ is $(\sqrt{m}M)$-Lipschitz continuous. For the weight matrix W, we make the following assumptions.

Assumption 3:

1) $W \in \mathbb{R}^{m\times m}$ is a symmetric matrix with $W_{i,j} \neq 0$ if and only if agents i and j are neighbors or $i = j$. Otherwise, $W_{i,j} = 0$.
2) $I \succeq W \succeq 0$, and $W\mathbf{1} = \mathbf{1}$.

Examples satisfying Assumption 3 can be found in [20]. We denote by $1 = \sigma_1(W) \geq \sigma_2(W) \geq \cdots \geq \sigma_m(W)$ the spectrum of W. Note that for a connected and undirected network, we always have $\sigma_2(W) < 1$, and $\frac{1}{1-\sigma_2(W)}$ is a good indication of the network connectivity. For many commonly used networks, we can give order-accurate estimate on $\frac{1}{1-\sigma_2(W)}$ [34, Proposition 5]. For example, $\frac{1}{1-\sigma_2(W)} = O(m\log m)$ for the geometric graph, and $\frac{1}{1-\sigma_2(W)} = O(1)$ for the expander graph and Erdős−Rényi random graph. Moreover, for any connected and undirected graph, $\frac{1}{1-\sigma_2(W)} = O(m^2)$ in the worst case [34].

In this paper, we focus on the communication and (sub)gradient computation complexity development for the proposed algorithms. We define one communication to be the operation that all the agents exchange information with their neighbors once, i.e., $\sum_{j\in\mathcal{N}_i} W_{ij}x_{(j)}$ for all $i = 1, 2, ..., m$. One (sub)gradient computation is defined to be the (sub)gradient evaluations of all the agents once, i.e., $\nabla f_i(x_{(i)})$ ($\hat{\nabla} h_i(x_{(i)})$) for all i.

II. Development of the Accelerated Penalty Method

A. Accelerated Penalty Method for Smooth Distributed Optimization

In this section, we consider the smooth distributed optimization, i.e., $h_i(x) = 0$ in problem (1). From the definition of Π in (3), we know that $x_{(1)} = \cdots = x_{(m)}$ is equivalent to $\Pi\mathbf{x} = 0$. Thus, we can reformulate the smooth distributed problem as

$$\min_{\mathbf{x}\in\mathbb{R}^{m\times n}} f(\mathbf{x}) \quad \text{s.t.} \quad \Pi\mathbf{x} = 0. \tag{4}$$

Problem (4) is a standard linearly constrained convex problem, and many algorithms can be used to solve it, e.g., the primal-dual method [13], [25], [35], [36] and dual ascent [9], [12], [23]. In order to propose an accelerated distributed gradient method based on the gradient of $f(\mathbf{x})$, rather than the evaluation of its Fenchel conjugate or proximal mapping, we follow [37] to use the penalty method to solve problem (4) in this paper. Specifically, the penalty method solves the following problem instead:

$$\min_{\mathbf{x}\in\mathbb{R}^{m\times n}} f(\mathbf{x}) + \frac{\beta}{2}\|\Pi\mathbf{x}\|_F^2, \tag{5}$$

where β is a large constant. However, one big issue of the penalty method is that problems (4) and (5) are not equivalent for finite β. When solving problem (5), we can only obtain

an approximate solution of (4) with small $\|\Pi\mathbf{x}\|_F$, rather than $\|\Pi\mathbf{x}\|_F \to 0$, and the algorithm only converges to a neighborhood of the solution set of problem (1) [37]. Moreover, to find an ϵ-optimal solution of (4), we need to pre-define a large β of the order $\frac{1}{\epsilon}$ [37]. Thus, the parameter setting depends on the precision ϵ. When β is fixed as a constant of the order $\frac{1}{\epsilon}$, we can only get the ϵ-accurate solution after some fixed iterations described by ϵ, and more iterations will not give a more accurate solution. Please see Section II-C1 for more details. To solve the above two problems, we use the gradually increasing penalty parameters, i.e., at the kth iteration, we use $\beta = \frac{\beta_0}{\vartheta_k}$ with fixed β_0 and diminishing $\vartheta_k \to 0$. The increasing penalty strategy has two advantages: 1) The solution of (5) approximates that of (4) infinitely when the iteration number k is sufficiently large. 2) The parameter setting does not depend on the accuracy ϵ. The algorithm can be run without defining the accuracy ϵ in advance. It can reach arbitrary accuracy if run for arbitrarily long time.

We use the classical accelerated proximal gradient method (APG) [38] to minimize the penalized objective in (5), i.e., at the kth iteration, we first compute the gradient of $f(\mathbf{x})$ at some extrapolated point, and then compute the proximal mapping of $\frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}\|_F^2$ at some $\mathbf{z}$, defined as

$$\operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^{m\times n}} \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}\|_F^2 + \frac{L}{2}\|\mathbf{x}-\mathbf{z}\|_F^2. \tag{6}$$

Due to the special form of Π defined in (3), a simple calculation yields $\frac{L\vartheta_k\mathbf{z}+\beta_0\mathbf{1}\alpha(\mathbf{z})^T}{L\vartheta_k+\beta_0}$ as the solution of (6), where $\alpha(\mathbf{x})$ is defined in (2). However, in the distributed setting, we can only compute $\alpha(\mathbf{z})$ approximately in finite communications. Thus, we use the inexact APG to minimize (5), i.e., we compute the proximal mapping inexactly. Specifically, the algorithm framework consists of the following steps:

$$\mathbf{y}^k = \mathbf{x}^k + \frac{L\theta_k-\mu}{L-\mu}\frac{1-\theta_{k-1}}{\theta_{k-1}}(\mathbf{x}^k-\mathbf{x}^{k-1}), \tag{7a}$$

$$\mathbf{z}^k = \mathbf{y}^k - \frac{1}{L}\nabla f(\mathbf{y}^k), \tag{7b}$$

$$\mathbf{x}^{k+1} \approx \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^{m\times n}} \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}\|_F^2 + \frac{L}{2}\left\|\mathbf{x}-\mathbf{z}^k\right\|_F^2, \tag{7c}$$

where the sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ and the precision in step (7c) will be specified in Theorems 1 and 2 latter. Now, we consider the subproblem in procedure (7c). As discussed above, we only need to approximate $\alpha(\mathbf{z}^k)$, which can be obtained by the classical average consensus [39] or the accelerated average consensus [40]. We only consider the accelerated average consensus, which consists of the following iterations:

$$\mathbf{z}^{k,t+1} = (1+\eta)W\mathbf{z}^{k,t} - \eta\mathbf{z}^{k,t-1}, \tag{8}$$

where we initialize at $\mathbf{z}^{k,0} = \mathbf{z}^{k,-1} = \mathbf{z}^k$. The advantage of using the special Π in (4) is that we only need to call the classic average consensus to solve the subproblem in (7c), which has been well studied in the literatures, including the extensions over directed network and time-varying network [41]. In fact, in Lemma 6, we only require $\|\mathbf{z}^{k,T_k}-\mathbf{1}\alpha(\mathbf{z}^k)\|_F^2$ to be within some precision for the method used in the inner loop. Any average consensus method over undirected graph, directed graph or time-varying graph can be used in the inner loop, as long as it has a linear convergence.

Algorithm 1 Accelerated Penalty Method with Consensus (APM-C)

Initialize $x_{(i)}^0 = x_{(i)}^{-1}$ for all i, and $\eta = \frac{1-\sqrt{1-\sigma_2^2(W)}}{1+\sqrt{1-\sigma_2^2(W)}}$.
for $k = 0, 1, 2, \cdots$ **do**
  $y_{(i)}^k = x_{(i)}^k + \frac{L\theta_k-\mu}{L-\mu}\frac{1-\theta_{k-1}}{\theta_{k-1}}\left(x_{(i)}^k - x_{(i)}^{k-1}\right) \quad \forall i,$
  $z_{(i)}^k = y_{(i)}^k - \frac{1}{L}\nabla f_i(y_{(i)}^k) \quad \forall i,$
  $z_{(i)}^{k,0} = z_{(i)}^{k,-1} = z_{(i)}^k \quad \forall i,$
  for $t = 0, 1, \cdots, T_k - 1$ **do**
    $z_{(i)}^{k,t+1} = (1+\eta)\sum_{j\in\mathcal{N}_i} W_{ij} z_{(j)}^{k,t} - \eta z_{(i)}^{k,t-1} \quad \forall i,$
  end for
  $x_{(i)}^{k+1} = \frac{L\vartheta_k z_{(i)}^k + \beta_0 z_{(i)}^{k,T_k}}{L\vartheta_k+\beta_0} \quad \forall i.$
end for

Combing (7a)-(7c) and (8), we can give our method, which is presented in a distributed way in Algorithm 1. We use notations $\mathbf{x}^{-1}$ and $\mathbf{z}^{k,-1}$ in Algorithm 1 only for the writing consistency when beginning the recursions from $k = 0$ and $t = 0$.

1) Complexities: In this section, we discuss the complexities of Algorithm 1. We first consider the strongly convex case and give the complexities in the following theorem.

Theorem 1: Assume that Assumptions 1 and 3 hold with $\mu > 0$. Setting $\theta_k = \theta = \sqrt{\frac{\mu}{L}}$ for all $\forall k$, $\vartheta_k = (1-\theta)^{k+1}$, and $T_k = O\left(\frac{k\sqrt{\mu/L}}{\sqrt{1-\sigma_2(W)}}\right)$. Then, Algorithm 1 needs $O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ gradient computations and $O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log^2\frac{1}{\epsilon}\right)$ total communications to achieve an ϵ-optimal solution $\mathbf{x}$ such that

$$\begin{aligned} &\frac{1}{m}\sum_{i=1}^m f_i(\alpha(\mathbf{x})) - \frac{1}{m}\sum_{i=1}^m f_i(x^*) \le \epsilon \\ &\frac{1}{m}\sum_{i=1}^m \left\|x_{(i)} - \alpha(\mathbf{x})\right\|^2 \le \epsilon^2. \end{aligned} \tag{9}$$

When we drop the strong-convexity assumption, we have the following theorem.

Theorem 2: Assume that Assumptions 1 and 3 hold with $\mu = 0$. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\theta_0 = 1$, $\frac{1-\theta_k}{\theta_k^2} = \frac{1}{\theta_{k-1}^2}$, and $\vartheta_k = \theta_k^2$. Setting $T_k = O\left(\frac{\log k}{\sqrt{1-\sigma_2(W)}}\right)$ and $\beta_0 \ge L\|\nabla f(\mathbf{x}^*)\|_F^2$. Then, Algorithm 1 needs $O\left(\sqrt{\frac{L}{\epsilon}}\right)$ gradient computations and $O\left(\sqrt{\frac{L}{\epsilon(1-\sigma_2(W))}}\log\frac{1}{\epsilon}\right)$ total communications to achieve an ϵ-optimal solution $\mathbf{x}$ such that (9) holds.

B. Accelerated Penalty Method for Nonsmooth Distributed Optimization

In this section, we consider the nonsmooth problem (1). From Assumption 3 and the definition in (3), we know $I \succeq U \succeq 0$, and $x_{(1)} = \cdots = x_{(m)}$ is equivalent to $U\mathbf{x} = 0$ [12]. Thus, similar to (4), we can reformulate problem (1) as

$$\min_{\mathbf{x}\in\mathbb{R}^{m\times n}} F(\mathbf{x}) \equiv f(\mathbf{x}) + h(\mathbf{x}) \quad \text{s.t.} \quad U\mathbf{x} = 0. \tag{10}$$

Similar to Section II-A, we also further rewrite the problem as a penalized problem and use APG with increasing penalties to minimize the penalized objective $F(\mathbf{x}) + \frac{\beta_0}{2\vartheta_k}\|U\mathbf{x}\|_F^2$. However, due to the nonsmooth term $h(\mathbf{x})$, we cannot compute the proximal mapping of $h(\mathbf{x}) + \frac{\beta_0}{2\vartheta_k}\|U\mathbf{x}\|_F^2$ efficiently. Thus, we use a slightly different strategy here. Specifically, we first compute the gradient of $f(\mathbf{x})+\frac{\beta_0}{2\vartheta_k}\|U\mathbf{x}\|_F^2$ at some extrapolated point $\mathbf{y}$, i.e., $\nabla f(\mathbf{y}) + \frac{\beta_0}{\vartheta_k}U^2\mathbf{y}$, and then compute the inexact proximal mapping of $h(\mathbf{x})$. We describe the iterations as follows:

$$\mathbf{y}^k = \mathbf{x}^k + \frac{\theta_k(1-\theta_{k-1})}{\theta_{k-1}}(\mathbf{x}^k - \mathbf{x}^{k-1}), \tag{11a}$$

$$\mathbf{s}^k = \nabla f(\mathbf{y}^k) + \frac{\beta_0}{\vartheta_k}U^2\mathbf{y}^k, \tag{11b}$$

$$\mathbf{x}^{k+1} \approx \underset{\mathbf{x}\in\mathbb{R}^{m\times n}}{\operatorname{argmin}}\, h(\mathbf{x}) + \left\langle \mathbf{s}^k, \mathbf{x}\right\rangle + \left(\frac{L}{2} + \frac{\beta_0}{2\vartheta_k}\right)\|\mathbf{x}-\mathbf{y}^k\|_F^2. \tag{11c}$$

The reason why we use U in (10), rather than Π, is that $U^2\mathbf{y}^k$ can be efficiently computed, which corresponds to the gossip-style communications. Otherwise, we need to compute the average across $y_{(1)}^k, ..., y_{(m)}^k$, which cannot be achieved with closed form solution in the distributed environment.

When the proximal mapping of $h(\mathbf{x})$, i.e., $\text{Prox}_h(\mathbf{z}) = \operatorname{argmin}_{\mathbf{x}\in\mathbb{R}^{m\times n}} h(\mathbf{x}) + \frac{1}{2}\|\mathbf{x}-\mathbf{z}\|^2$ for some $\mathbf{z}$, has closed form solution or can be easily computed, step (11c) has a low computation cost, which reduces to

$$\mathbf{x}^{k+1} = \text{Prox}_h\left(\mathbf{y}^k - \frac{1}{L+\beta_0/\vartheta_k}\left(\nabla f(\mathbf{y}^k) + \frac{\beta_0}{\vartheta_k}U^2\mathbf{y}^k\right)\right). \tag{12}$$

We can see that when we set a large penalty parameter β, i.e., exchange $\frac{\beta_0}{\vartheta_k}$ with a large β such that $\beta \gg L$ in (12), (12) approximately reduces to $\mathbf{x}^{k+1} \approx \text{Prox}_h\left(\mathbf{y}^k - U^2\mathbf{y}^k\right)$ and $\nabla f(\mathbf{y}^k)$ is flooded by the large penalty parameters. This is another reason to use the increasing penalty parameters.

When the proximal mapping of $h(\mathbf{x})$ does not have a closed form solution, we borrow the idea of gradient and communication sliding proposed in [25], [42]–[44], which skips the computations of ∇f and the inter-node communications from time to time so that only $O(1/\epsilon)$ gradient evaluations and communications are needed in the $O(1/\epsilon^2)$ iterations required to solve problem (10). Specifically, we incorporate a subgradient descent procedure to solve the subproblem in (11c) with a sliding period T_k, which is also adopted by [13]. The subgradient descent is described as follows for T_k iterations:

$$\mathbf{z}^{k,t+1} = \underset{\mathbf{z}\in\mathbb{R}^{m\times n}}{\operatorname{argmin}} \left\langle \hat{\nabla} h(\mathbf{z}^{k,t}), \mathbf{z}\right\rangle + \left\langle \mathbf{s}^k, \mathbf{z}\right\rangle + \left(\frac{L}{2} + \frac{\beta_0}{2\vartheta_k}\right)\|\mathbf{z}-\mathbf{y}^k\|_F^2 + \frac{1}{2\eta_k}\|\mathbf{z}-\mathbf{z}^{k,t}\|_F^2.$$

We describe the method in a distributed way in Algorithm 2.

1) Complexities: Introduce constants R_1 and R_2 such that

$$\|x_{(i)}^0 - x^*\|^2 \le R_1^2 \quad \text{and} \quad \|\nabla f_i(x^*)\|^2 \le R_2^2 \quad \text{for all } i, \tag{13}$$

and assume $R_1 \ge 1$ for simplicity. Then, we describe the convergence rate for Algorithm 2 in the following theorem.

Theorem 3: Assume that Assumptions 1, 2 and 3 hold with $\mu = 0$. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\theta_0 = 1$, $\frac{1-\theta_k}{\theta_k} = \frac{1}{\theta_{k-1}}$, and $\vartheta_k = \theta_k$. Set $T_k = K(1-\sigma_2(W))$, $\eta_k = \frac{\theta_k}{KM\sqrt{1-\sigma_2(W)}}$, and $\beta_0 = \frac{\max\{M,L\}}{\sqrt{1-\sigma_2(W)}}$, where K is the number of outer iterations. Then, for Algorithm 2, we have

$$\frac{1}{m}\sum_{i=1}^m F_i\left(\alpha(\mathbf{x}^K)\right) - \frac{1}{m}\sum_{i=1}^m F_i(x^*) \le \frac{\beta_0}{K}\left(31 + \frac{8}{K\sqrt{1-\sigma_2(W)}}\right)\left(R_1 + \frac{R_2}{L}\right)^2,$$

and

$$\frac{1}{m}\sum_{i=1}^m \left\|x_{(i)}^K - \alpha(\mathbf{x}^K)\right\|^2 \le \frac{16\beta_0^2}{K^2M^2}\left(R_1 + \frac{R_2}{L}\right)^2.$$

Algorithm 2 Accelerated Penalty Method (APM)

Initialize $x_{(i)}^0 = x_{(i)}^{-1}$, and $z_{(i)}^{-1,T_{-1}} = x_{(i)}^0$ for all i.
for $k = 0, 1, 2, \cdots, K$ **do**
  $y_{(i)}^k = x_{(i)}^k + \frac{\theta_k(1-\theta_{k-1})}{\theta_{k-1}}(x_{(i)}^k - x_{(i)}^{k-1}) \quad \forall i,$
  $\mathbf{s}_{(i)}^k = \nabla f_i(y_{(i)}^k) + \frac{\beta_0}{\vartheta_k}\left(y_{(i)}^k - \sum_{j=1}^m W_{i,j}y_{(j)}^k\right) \quad \forall i,$
  $z_{(i)}^{k,0} = z_{(i)}^{k-1,T_{k-1}} \quad \forall i,$
  for $t = 0, 1, \cdots, T_k - 1$ **do**
    $z_{(i)}^{k,t+1} = \operatorname{argmin}_{z\in\mathbb{R}^n}\left\langle \hat{\nabla} h_i(z_{(i)}^{k,t}) + \mathbf{s}_{(i)}^k, z\right\rangle + \left(\frac{L}{2} + \frac{\beta_0}{2\vartheta_k}\right)\|z - y_{(i)}^k\|^2 + \frac{1}{2\eta_k}\|z - z_{(i)}^{k,t}\|^2 \quad \forall i.$
  end for
  $x_{(i)}^{k+1} = \frac{\sum_{t=0}^{T_k-1} z_{(i)}^{k,t+1}}{T_k} \quad \forall i.$
end for

Consider the simple problem of computing the average of $x_{(1)}, \cdots, x_{(m)}$. The accelerated averaged consensus [40] needs $O\left(\frac{1}{\sqrt{1-\sigma_2(W)}}\log\frac{1}{\epsilon}\right)$ iterations to find an ϵ-accurate solution. Thus, it is reasonable to assume $K \ge \frac{1}{\sqrt{1-\sigma_2(W)}}$. Moreover, from the L-smoothness of $f_i(x)$, we know R_2 is often of the order $O(LR_1)$. Thus, Theorem 3 establishes the $O\left(\frac{\max\{M,L\}}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$ communication complexity and the $\sum_{k=1}^K T_k = K^2(1-\sigma_2(W)) = O\left(\frac{\max\{M,L\}^2}{\epsilon^2}\right)$ subgradient computation complexity such that (9) holds for nonsmooth distributed optimization.

In Theorem 3, we set T_k and η_k dependent on the number of outer iterations. As explained in Section II-A, it is a unpractical parameter setting and moreover, the large T_k and $\frac{1}{\eta_k}$ make the algorithm slow in practice. In the following corollary, we give a more reasonable setting of the parameters at an expense of higher complexities by the order of $\log\frac{1}{\epsilon}$, i.e., $\log K$.

Corollary 1: Under the settings in Theorem 3 but with $T_k = \frac{1-\sigma_2(W)}{\theta_k}$ and $\eta_k = \frac{\theta_k^2}{M\sqrt{1-\sigma_2(W)}}$, we have

$$\frac{1}{m}\sum_{i=1}^m F_i\left(\alpha(\mathbf{x}^K)\right) - \frac{1}{m}\sum_{i=1}^m F_i(x^*) \le \frac{\beta_0\log K}{K}\left(31 + \frac{8}{K\sqrt{1-\sigma_2(W)}}\right)\left(R_1 + \frac{R_2}{L}\right)^2,$$

and

$$\frac{1}{m}\sum_{i=1}^{m}\left\|x_{(i)}^K-\alpha(\mathbf{x}^K)\right\|^2\le\frac{16\beta_0^2\log K}{K^2M^2}\left(R_1+\frac{R_2}{L}\right)^2.$$

When $f(\mathbf{x})$ is strongly convex, we can prove a faster $O\left(\frac{1}{k^2}\right)$ convergence rate for Algorithm 2 with $\theta_k=\frac{2}{k+2}$ and $\vartheta_k=\theta_k^2$. However, the quickly diminishing step-size in step (11c) makes the algorithm slow in practice. So we omit the discussion for the strongly convex case.

C. Relations of APM-C and APM to the Existing Algorithm Frameworks

1) Difference from the classical penalty method: To the best of our knowledge, most traditional work analyze the penalty method with a fixed penalty parameter [45], [46]. Let's discuss the disadvantage of the large and fixed penalty parameter. Take problem (4) as an example. Let $\{\mathbf{x}^*,\lambda^*\}$ be a pari of KKT point of problem (4) and $\hat{\mathbf{x}}^*$ be the minimizer of problem (5), from the proof in [45, Proposition 10], we have

$$f(\mathbf{x}^*)=f(\mathbf{x}^*)+\frac{\beta}{2}\|\Pi\mathbf{x}^*\|_F^2\ge f(\hat{\mathbf{x}}^*)+\frac{\beta}{2}\|\Pi\hat{\mathbf{x}}^*\|_F^2.$$

So for any ε-accurate solution $\mathbf{x}$ of problem (5), we have

$$\begin{aligned}&f(\mathbf{x})+\frac{\beta}{2}\|\Pi\mathbf{x}\|_F^2-f(\mathbf{x}^*)\\ \le&f(\mathbf{x})+\frac{\beta}{2}\|\Pi\mathbf{x}\|_F^2-f(\hat{\mathbf{x}}^*)-\frac{\beta}{2}\|\Pi\hat{\mathbf{x}}^*\|_F^2\le\varepsilon.\end{aligned}$$

On the other hand, since $\mathbf{x}^*=\operatorname{argmin}_{\mathbf{x}}f(\mathbf{x})+\langle\lambda^*,\Pi\mathbf{x}\rangle$ and $\Pi\mathbf{x}^*=0$, we have

$$\begin{aligned}&f(\mathbf{x}^*)=f(\mathbf{x}^*)+\langle\lambda^*,\Pi\mathbf{x}^*\rangle\le f(\mathbf{x})+\langle\lambda^*,\Pi\mathbf{x}\rangle\\ \Rightarrow&-\|\lambda^*\|_F\|\Pi\mathbf{x}\|_F\le f(\mathbf{x})-f(\mathbf{x}^*).\end{aligned}$$

So $\frac{\beta}{2}\|\Pi\mathbf{x}\|_F^2-\|\lambda^*\|_F\|\Pi\mathbf{x}\|_F\le\varepsilon$, which leads to

$$\|\Pi\mathbf{x}\|_F\le\frac{2\|\lambda^*\|_F}{\beta}+\sqrt{\frac{2\varepsilon}{\beta}}=O(\epsilon+\sqrt{\epsilon\varepsilon})$$

and

$$|f(\mathbf{x})-f(\mathbf{x}^*)|\le\max\{\varepsilon,\epsilon+\sqrt{\epsilon\varepsilon}\}$$

by $\beta=\frac{1}{\epsilon}$. We can see that the accuracy is dominated by $\max\{\epsilon,\varepsilon\}$, and more iterations with smaller ε will not produce a more accurate solution.

On the other hand, even if $\varepsilon=0$ and $\mathbf{x}=\hat{\mathbf{x}}^*$ with infinite iterations, we have $\nabla f(\mathbf{x})+\beta\Pi\mathbf{x}=0$, which only leads to $\|\Pi\mathbf{x}\|_F=\epsilon\|\nabla f(\mathbf{x})\|_F=O(\epsilon)$ and $|f(\mathbf{x})-f(\mathbf{x}^*)|\le\epsilon$, rather than $\|\Pi\mathbf{x}\|_F=0$ and $|f(\mathbf{x})-f(\mathbf{x}^*)|=0$.

2) Difference from the classical accelerated first-order algorithms: We extend the classical accelerated gradient method [38], [47]–[50] from the unconstrained problems to the linearly constrained problems via the perspective of the penalty method. However, since we use the increasing penalty parameters at each iteration, i.e., the penalized objective varies at different iterations, the conclusion in [38], [49], [50] for the unconstrained problems cannot be directly used for procedures (7a)-(7c) and (11a)-(11c). The increasing penalty parameters make the convergence analysis more challenging.

3) Difference from the accelerated gradient sliding method: [9] combined Nesterov's smoothing technique [51] with the accelerated gradient sliding methods [42]–[44] to solve the nonsmooth problem (10) with $f(\mathbf{x})=0$. In fact, when fixing the penalty parameter as a large one of the order $O(1/\epsilon)$, Algorithm 2 is similar to the one in [9, Section 6.3]. However, our method adopts increasing penalty parameters such that it avoids having to set a large inner iteration number T_k and a small step-size η_k at the beginning of the outer loop, as shown in Corollary 1. On the other hand, when $f(\mathbf{x})\neq0$, as explained in Section II-B, $\nabla f(\mathbf{y}^k)$ is flooded if we set a large and fixed penalty parameter.

4) Difference from the D-NC and D-NG in [8]: Algorithm 1 can be seen as an improvement over the D-NC proposed in [8]. Both Algorithm 1 and D-NC use Nesterov's acceleration technique and multi-consensus, and both attain the optimal computation complexity for the nonstrongly convex problems. However, Algorithm 1 is motivated by a constraint-penalty approach while D-NC is developed from the inexact accelerated gradient method [49] directly. Moreover, Algorithm 1 can solve both the strongly convex and nonstrongly convex problems while [8] only studied the nonstrongly convex case.

As for Algorithm 2, consider the simple case with $h(\mathbf{x})=0$ and $\frac{\beta_0}{\vartheta_k}=\frac{k+1}{c}$, then steps (11b) and (11c) become

$$\mathbf{x}^{k+1}=\frac{L\mathbf{y}^k+(k+1)W\mathbf{y}^k/c}{L+(k+1)/c}-\frac{\nabla f(\mathbf{y}^k)}{L+(k+1)/c}.$$

Thus, when $(k+1)/c\gg L$, we have $\mathbf{x}^{k+1}\approx W\mathbf{y}^k-\frac{c}{k+1}\nabla f(\mathbf{y}^k)$ and it approximates the D-NG in [8]. Algorithm 2 gives a different explanation of the D-NG, and it improves the D-NG in the sense that it handles a possible nondifferentiable function $h_i(\mathbf{x})$. The complexity of D-NG is $O\left(\frac{1}{\epsilon(1-\sigma_2(W))^{1+\xi}}\log\frac{1}{\epsilon}\right)$, where ξ is a small constant. Our complexity, i.e., $O\left(\frac{1}{\epsilon\sqrt{1-\sigma_2(W)}}\right)$, is better because theirs has the extra $\log\frac{1}{\epsilon}$ factor and is more sensitive to $1-\sigma_2(W)$.

III. Proof of Theorems

A. Supporting Lemmas

Before providing a comprehensive convergence analysis for Algorithms 1 and 2, we first present some useful technical lemmas. We first give the following easy-to-identify identifies.

Lemma 1: For any $\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{w}\in\mathbb{R}^{m\times n}$, we have the following two identities:

$$\begin{aligned}2\langle\mathbf{x}-\mathbf{z},\mathbf{y}-\mathbf{z}\rangle&=\|\mathbf{x}-\mathbf{z}\|_F^2+\|\mathbf{y}-\mathbf{z}\|_F^2-\|\mathbf{x}-\mathbf{y}\|_F^2,\\ 2\langle\mathbf{x}-\mathbf{z},\mathbf{y}-\mathbf{w}\rangle&=\|\mathbf{y}-\mathbf{z}\|_F^2-\|\mathbf{w}-\mathbf{z}\|_F^2+\|\mathbf{x}-\mathbf{w}\|_F^2-\|\mathbf{x}-\mathbf{y}\|_F^2.\end{aligned}$$

In the following Lemma, we bound the Lagrange multiplier, which is useful for the complexity analysis in the distributed optimization community.

Lemma 2: Assume that Assumptions 1, 2 and 3 hold with $\mu\ge0$. Then, we have the following properties:

1) There exists a pair of KKT points $(\mathbf{x}^*,\lambda^*)$ of saddle point problem $\min_{\mathbf{x}}\max_{\lambda}f(\mathbf{x})+\langle\lambda,\Pi\mathbf{x}\rangle$, such that $\|\lambda^*\|_F\le\|\nabla f(\mathbf{x}^*)\|_F$.

2) There exists a pair of KKT points $(\mathbf{x}^*, \lambda^*)$ of saddle point problem $\min_{\mathbf{x}} \max_{\lambda} f(\mathbf{x}) + \langle \lambda, U\mathbf{x}\rangle$, such that $\|\lambda^*\|_F \leq \frac{\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}$.
3) There exists a pair of KKT points $(\mathbf{x}^*, \lambda^*)$ of saddle point problem $\min_{\mathbf{x}} \max_{\lambda} F(\mathbf{x}) + \langle \lambda, U\mathbf{x}\rangle$, such that $\|\lambda^*\|_F \leq \frac{\sqrt{m}M+\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}$.

The proof can be found in [25, Theorem 2]. The following lemma is a corollary of the saddle point property.

Lemma 3: [52] If $f(\mathbf{x})$ is convex and $(\mathbf{x}^*, \lambda^*)$ is a pair of KKT points of saddle point problem $\min_{\mathbf{x}} \max_{\lambda} f(\mathbf{x}) + \langle \lambda, A\mathbf{x}\rangle$, then we have $f(\mathbf{x}) - f(\mathbf{x}^*) + \langle \lambda^*, A\mathbf{x}\rangle \geq 0$ for all $\mathbf{x}$.

The following lemma bounds the consensus violation of $\|\Pi\mathbf{x}\|_F$ from $\|U\mathbf{x}\|_F$.

Lemma 4: Assume that Assumption 3 holds. Then, we have $\|\Pi\mathbf{x}\|_F \leq \frac{1}{\sqrt{1-\sigma_2(W)}}\|U\mathbf{x}\|_F$.

Proof 1: From Assumption 3, we know $U\mathbf{1} = 0$, $U = U^T$, and $\text{rank}(U) = m-1$. For any $\mathbf{x} \in \mathbb{R}^{m\times n}$, denote $\overline{\mathbf{x}} = \Pi\mathbf{x} = \mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}$. Since $\mathbf{1}^T\overline{\mathbf{x}} = 0$, we know $\overline{\mathbf{x}}$ is orthogonal to the null space of U, and thus it belongs to the row (i.e., column) space of U. Let $V\Sigma V^T = U$ be its economical SVD with $V \in \mathbb{R}^{m\times(m-1)}$. Then we have

$$\begin{aligned}
&\|U\mathbf{x}\|_F^2 = \|U\overline{\mathbf{x}}\|_F^2 = \sum_{i=1}^n \overline{\mathbf{x}}_i^T U^2 \overline{\mathbf{x}}_i = \sum_{i=1}^n (V^T\overline{\mathbf{x}}_i)^T\Sigma^2(V^T\overline{\mathbf{x}}_i)\\
&\geq (1-\sigma_2(W))\sum_{i=1}^n \|V^T\overline{\mathbf{x}}_i\|_F^2 = (1-\sigma_2(W))\|V^T\overline{\mathbf{x}}\|_F^2\\
&\overset{a}{=} (1-\sigma_2(W))\|\overline{\mathbf{x}}\|_F^2 = (1-\sigma_2(W))\|\Pi\mathbf{x}\|_F^2,
\end{aligned}$$

where we denote $\mathbf{x}_i$ to be the ith column of $\mathbf{x}$, and $\overset{a}{=}$ follows from the fact that $\overline{\mathbf{x}}$ belongs to the column space of U, i.e., there exists $\alpha \in \mathbb{R}^{(m-1)\times n}$ such tht $\overline{\mathbf{x}} = V\alpha$. □

At last, we present the following lemma, which can be used to analyze the algorithms with inexact subproblem computation.

Lemma 5: [50] Assume that (s_k) is a sequence with increasing scalars and (v_k), (α_i) are sequences with nonnegative scalars, $v_0^2 \leq s_0$. If $v_k^2 \leq s_k + \sum_{i=1}^k \alpha_i v_i$, then we have $v_k \leq \frac{1}{2}\sum_{i=1}^k \alpha_i + \sqrt{\left(\frac{1}{2}\sum_{i=1}^k \alpha_i\right)^2 + s_k}$.

B. Complexity Analysis for Algorithm 1

1) Inner Loop: Before proving the convergence of procedure (7a)-(7c), we first establish the required precision to approximate $\alpha(\mathbf{z}^k)$ for an ε_k-accurate solution of the subproblem in (7c).

Lemma 6: Let $\mathbf{z}^{k,T_k}$ be obtained by (8) and $\mathbf{x}^{k+1} = \frac{L\vartheta_k\mathbf{z}^k + \beta_0\mathbf{z}^{k,T_k}}{L\vartheta_k+\beta_0}$. If $\|\mathbf{z}^{k,T_k} - \mathbf{1}\alpha(\mathbf{z}^k)^T\|_F^2 \leq \frac{2\vartheta_k\varepsilon_k}{\beta_0}$, then we have

$$\begin{aligned}
&\frac{L}{2}\left\|\mathbf{x}^{k+1} - \mathbf{z}^k\right\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}^{k+1}\|_F^2\\
&\leq \min_{\mathbf{x}\in\mathbb{R}^{m\times n}} \frac{L}{2}\left\|\mathbf{x} - \mathbf{z}^k\right\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}\|_F^2 + \varepsilon_k.
\end{aligned} \tag{14}$$

Proof 2: Define $\mathbf{x}^{k,*} = \text{argmin}_{\mathbf{x}} \frac{L}{2}\|\mathbf{x}-\mathbf{z}^k\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}\|_F^2$, $\widetilde{\mathbf{x}}^{k,*} = \frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{x}^{k,*}$, and $\widetilde{\mathbf{z}}^k = \frac{1}{m}\mathbf{1}\mathbf{1}^T\mathbf{z}^k$. From the optimality condition of $\mathbf{x}^{k,*}$, we have

$$0 = L(\mathbf{x}^{k,*} - \mathbf{z}^k) + \frac{\beta_0}{\vartheta_k}\Pi^2\mathbf{x}^{k,*}. \tag{15}$$

From $\Pi = \Pi^2$ and its definition, we have $0 = L(\mathbf{x}^{k,*} - \mathbf{z}^k) + \frac{\beta_0}{\vartheta_k}(\mathbf{x}^{k,*} - \widetilde{\mathbf{x}}^{k,*})$, which leads to $\mathbf{x}^{k,*} = \frac{L\vartheta_k\mathbf{z}^k + \beta_0\widetilde{\mathbf{x}}^{k,*}}{L\vartheta_k+\beta_0}$. Multiplying both sides of (15) by $\frac{1}{m}\mathbf{1}\mathbf{1}^T$, and using $\mathbf{1}^T\Pi = 0$, we have $\widetilde{\mathbf{x}}^{k,*} = \widetilde{\mathbf{z}}^k$, which further gives $\mathbf{x}^{k,*} = \frac{L\vartheta_k\mathbf{z}^k + \beta_0\widetilde{\mathbf{z}}^k}{L\vartheta_k+\beta_0}$. On the other hand, we have

$$\begin{aligned}
&\frac{L}{2}\left\|\mathbf{x}^{k+1} - \mathbf{z}^k\right\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}^{k+1}\|_F^2\\
&- \frac{L}{2}\left\|\mathbf{x}^{k,*} - \mathbf{z}^k\right\|_F^2 - \frac{\beta_0}{2\vartheta_k}\|\Pi\mathbf{x}^{k,*}\|_F^2\\
&\overset{a}{=} L\left\langle \mathbf{x}^{k,*} - \mathbf{z}^k, \mathbf{x}^{k+1} - \mathbf{x}^{k,*}\right\rangle + \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k,*}\|_F^2\\
&+ \frac{\beta_0}{\vartheta_k}\left\langle \Pi^2\mathbf{x}^{k,*}, \mathbf{x}^{k+1} - \mathbf{x}^{k,*}\right\rangle + \frac{\beta_0}{2\vartheta_k}\|\Pi(\mathbf{x}^{k+1} - \mathbf{x}^{k,*})\|_F^2\\
&\overset{b}{=} \frac{L}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k,*}\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi(\mathbf{x}^{k+1} - \mathbf{x}^{k,*})\|_F^2\\
&\overset{c}{=} \frac{\beta_0^2}{(L\vartheta_k+\beta_0)^2}\left(\frac{L}{2}\|\mathbf{z}^{k,T_k} - \widetilde{\mathbf{z}}^k\|_F^2 + \frac{\beta_0}{2\vartheta_k}\|\Pi(\mathbf{z}^{k,T_k} - \widetilde{\mathbf{z}}^k)\|_F^2\right)\\
&\overset{d}{\leq} \frac{\beta_0^2}{2\vartheta_k(L\vartheta_k+\beta_0)}\|\mathbf{z}^{k,T_k} - \widetilde{\mathbf{z}}^k\|_F^2 \leq \frac{\beta_0}{2\vartheta_k}\|\mathbf{z}^{k,T_k} - \widetilde{\mathbf{z}}^k\|_F^2.
\end{aligned} \tag{16}$$

where we use Lemma 1 in $\overset{a}{=}$, (15) in $\overset{b}{=}$, the definition of $\mathbf{x}^{k+1}$ and $\mathbf{x}^{k,*} = \frac{L\vartheta_k\mathbf{z}^k + \beta_0\widetilde{\mathbf{z}}^k}{L\vartheta_k+\beta_0}$ in $\overset{c}{=}$, and $\|\Pi\mathbf{z}\|_F \leq \|\mathbf{z}\|_F$ in $\overset{d}{\leq}$. □

Now we consider the iteration number of the accelerated average consensus in (8) to solve the subproblem in (7c) such that (14) is satisfied. From [40, Proposition 3], we have

$$\begin{aligned}
\|\mathbf{z}^{k,T_k} - \mathbf{1}\alpha(\mathbf{z}^k)^T\|_F &\leq \left(\frac{\sigma_2(W)}{1+\sqrt{1-\sigma_2^2(W)}}\right)^{T_k}\|\Pi\mathbf{z}^k\|_F \qquad (17)\\
&\leq \left(\frac{\sigma_2(W)}{1+\sqrt{1-\sigma_2(W)}}\right)^{T_k}\|\Pi\mathbf{z}^k\|_F = \left(1-\sqrt{1-\sigma_2(W)}\right)^{T_k}\|\Pi\mathbf{z}^k\|_F.
\end{aligned}$$

Thus, from Lemma 6, we only need

$$T_k = \frac{1}{-2\log\left(1-\sqrt{1-\sigma_2(W)}\right)}\log\frac{\beta_0\|\Pi\mathbf{z}^k\|_F^2}{2\vartheta_k\varepsilon_k} \tag{18}$$

such that (14) is satisfied.

At last, we study the property when the proximal mapping in (7c) is inexactly computed. When it is computed exactly, i.e., $\varepsilon_k = 0$ in (14), we have $L(\mathbf{x}^{k+1} - \mathbf{z}^k) + \frac{\beta_0}{\vartheta_k}\Pi^2\mathbf{x}^{k+1} = 0$. However, when it is computed inexactly, we should modify the conclusion accordingly. Specifically, we give the following lemma.

Lemma 7: Assume that (14) holds. Then, there exists δ^k with $\|\delta^k\|_F \leq \sqrt{\frac{2\varepsilon_k}{L}}$ and $\frac{\beta_0}{\vartheta_k}\|\Pi\delta^k\|_F^2 \leq 2\varepsilon_k$ such that

$$L(\mathbf{x}^{k+1} - \mathbf{z}^k + \delta^k) + \frac{\beta_0}{\vartheta_k}\Pi^2(\mathbf{x}^{k+1} + \delta^k) = 0. \tag{19}$$

Proof 3: Define $\delta^k = \mathbf{x}^{k,*} - \mathbf{x}^{k+1}$. From (14) and equation $\overset{b}{=}$ in (16), we have $\|\delta^k\|_F \leq \sqrt{\frac{2\varepsilon_k}{L}}$ and $\frac{\beta_0}{\vartheta_k}\|\Pi\delta^k\|_F^2 \leq 2\varepsilon_k$. From (15) and the definition of δ^k, we have (19). □

2) Outer Loop: Now we are ready to analyze procedure (7a)-(7c). Define

$$\mathbf{w}^{k+1} \equiv \frac{\mathbf{x}^{k+1}}{\theta_k} - \frac{1-\theta_k}{\theta_k}\mathbf{x}^k \text{ for any } k \geq 0 \text{ and } \mathbf{w}^0 = \mathbf{x}^0.$$

From the definition of $\mathbf{y}^k$ in (7a), we can give the following easy-to-identify identities.

Lemma 8: For procedure (7a)-(7c), we have

$$\begin{aligned}&\mathbf{x}^*+\frac{(1-\theta_k)L}{L\theta_k-\mu}\mathbf{x}^k-\frac{L-\mu}{L\theta_k-\mu}\mathbf{y}^k=\mathbf{x}^*-\mathbf{w}^k,\\&\theta_k\mathbf{x}^*+(1-\theta_k)\mathbf{x}^k-\mathbf{x}^{k+1}=\theta_k\left(\mathbf{x}^*-\mathbf{w}^{k+1}\right).\end{aligned}$$

Let $(\mathbf{x}^*,\lambda^*)$ be a pair of KKT points of saddle point problem $\min_{\mathbf{x}}\max_{\lambda} f(\mathbf{x})+\langle\lambda,\Pi\mathbf{x}\rangle$ satisfying Lemma 2. Define

$$\rho_{k+1}=f(\mathbf{x}^{k+1})-f(\mathbf{x}^*)+\left\langle\lambda^*,\Pi\mathbf{x}^{k+1}\right\rangle. \tag{20}$$

From Lemma 3, we know $\rho_{k+1}\geq 0$.

We first give the following lemma, which describes a progress in one iteration of procedure (7a)-(7c).

Lemma 9: Assume that Assumption 1 holds with $\mu\geq 0$. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\frac{1-\theta_k}{\vartheta_k}=\frac{1}{\vartheta_{k-1}}$ and $\theta_k\geq\frac{\mu}{L}$. Then, under the assumption of (14), we have

$$\begin{aligned}&\rho_{k+1}+\frac{\vartheta_k}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1}-\lambda^*\right\|_F^2+\frac{L\theta_k^2}{2}\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F^2\\\leq&(1-\theta_k)\rho_k+\frac{\vartheta_k}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_{k-1}}\Pi\mathbf{x}^k-\lambda^*\right\|_F^2+\varepsilon_k\\&+\frac{(L\theta_k-\mu)\theta_k}{2}\left\|\mathbf{w}^k-\mathbf{x}^*\right\|_F^2+L\theta_k\sqrt{\frac{2\varepsilon_k}{L}}\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F.\end{aligned} \tag{21}$$

Proof 4: From the smoothness and convexity of $f(\mathbf{x})$, we have

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

Plugging $\mathbf{z}^k=\mathbf{y}^k-\frac{1}{L}\nabla f(\mathbf{y}^k)$ and (19) into the above inequality, we have

$$\begin{aligned}&f(\mathbf{x}^{k+1})-f(\mathbf{x})\\\leq&\frac{\beta_0}{\vartheta_k}\left\langle\Pi\mathbf{x}^{k+1}+\Pi\delta^k,\Pi\mathbf{x}-\Pi\mathbf{x}^{k+1}\right\rangle+L\left\langle\mathbf{x}^{k+1}-\mathbf{y}^k,\mathbf{x}-\mathbf{y}^k\right\rangle\\&+L\left\langle\delta^k,\mathbf{x}-\mathbf{x}^{k+1}\right\rangle-\frac{\mu}{2}\|\mathbf{x}-\mathbf{y}^k\|_F^2-\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2\end{aligned}$$

When we apply (23) first with $\mathbf{x}=\mathbf{x}^k$ and then with $\mathbf{x}=\mathbf{x}^*$, we obtain two inequalities. Multiplying the first inequality by $(1-\theta_k)$, multiplying the second by θ_k, adding them together with $\left\langle\lambda^*,\Pi\mathbf{x}^{k+1}-(1-\theta_k)\Pi\mathbf{x}^k\right\rangle$ to both sides, and using $\Pi\mathbf{x}^*=0$, we have

$$\begin{aligned}&f(\mathbf{x}^{k+1})-(1-\theta_k)f(\mathbf{x}^k)-\theta_k f(\mathbf{x}^*)\\&+\left\langle\lambda^*,\Pi\mathbf{x}^{k+1}-(1-\theta_k)\Pi\mathbf{x}^k\right\rangle\\\leq&\left\langle\frac{\beta_0}{\vartheta_k}(\Pi\mathbf{x}^{k+1}+\Pi\delta^k)-\lambda^*,(1-\theta_k)\Pi\mathbf{x}^k-\Pi\mathbf{x}^{k+1}\right\rangle\\&+L\left\langle\mathbf{x}^{k+1}-\mathbf{y}^k,(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{y}^k\right\rangle\\&+L\left\langle\delta^k,(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{x}^{k+1}\right\rangle\\&-\frac{\mu\theta_k}{2}\|\mathbf{x}^*-\mathbf{y}^k\|_F^2-\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2\\\overset{a}{=}&\frac{\vartheta_k}{\beta_0}\left\langle\frac{\beta_0}{\vartheta_k}(\Pi\mathbf{x}^{k+1}+\Pi\delta^k)-\lambda^*,\frac{\beta_0}{\vartheta_{k-1}}\Pi\mathbf{x}^k-\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1}\right\rangle\\&+L\left\langle\mathbf{x}^{k+1}-\mathbf{y}^k,(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{y}^k\right\rangle\\&+L\left\langle\delta^k,(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{x}^{k+1}\right\rangle\\&-\frac{\mu\theta_k}{2}\|\mathbf{x}^*-\mathbf{y}^k\|_F^2-\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2,\end{aligned}$$

where we use $\frac{1-\theta_k}{\vartheta_k}=\frac{1}{\vartheta_{k-1}}$ in $\overset{a}{=}$. Applying the identities in Lemma 1 to the two inner products, we have

$$\begin{aligned}&\rho_{k+1}-(1-\theta_k)\rho_k\\\leq&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}\Pi\mathbf{x}^k-\lambda^*\right\|_F^2+\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1}-\frac{\beta_0}{\vartheta_k}(\Pi\mathbf{x}^{k+1}+\Pi\delta^k)\right\|_F^2\right.\\&\left.-\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1}-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_{k-1}}\Pi\mathbf{x}^k-\frac{\beta_0}{\vartheta_k}(\Pi\mathbf{x}^{k+1}+\Pi\delta^k)\right\|_F^2\right]\\&+\frac{L}{2}\left[\|(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{y}^k\|_F^2-\|(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{x}^{k+1}\|_F^2\right]\\&+L\left\langle\delta^k,(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*-\mathbf{x}^{k+1}\right\rangle-\frac{\mu\theta_k}{2}\|\mathbf{x}^*-\mathbf{y}^k\|_F^2\\\overset{b}{\leq}&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}\Pi\mathbf{x}^k-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1}-\lambda^*\right\|_F^2+\frac{\beta_0^2}{\vartheta_k^2}\|\Pi\delta^k\|_F^2\right]\\&+\frac{L\theta_k^2}{2}\left[\left\|\frac{\mathbf{y}^k}{\theta_k}-\frac{1-\theta_k}{\theta_k}\mathbf{x}^k-\mathbf{x}^*\right\|_F^2-\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F^2\right]\\&-L\theta_k\left\langle\delta^k,\mathbf{w}^{k+1}-\mathbf{x}^*\right\rangle-\frac{\mu\theta_k}{2}\|\mathbf{x}^*-\mathbf{y}^k\|_F^2.\end{aligned} \tag{22}$$

where $\overset{b}{\leq}$ follows from the second identity in Lemma 8. By reorganizing the terms in $\frac{\mathbf{y}^k}{\theta_k}-\frac{1-\theta_k}{\theta_k}\mathbf{x}^k-\mathbf{x}^*$ carefully, we have

$$\begin{aligned}&\frac{L\theta_k^2}{2}\left\|\frac{\mathbf{y}^k}{\theta_k}-\frac{1-\theta_k}{\theta_k}\mathbf{x}^k-\mathbf{x}^*\right\|_F^2\\=&\frac{L\theta_k^2}{2}\left\|\frac{\mu}{L\theta_k}(\mathbf{y}^k-\mathbf{x}^*)+\left(1-\frac{\mu}{L\theta_k}\right)\left(\frac{L-\mu}{L\theta_k-\mu}\mathbf{y}^k-\frac{(1-\theta_k)L}{L\theta_k-\mu}\mathbf{x}^k-\mathbf{x}^*\right)\right\|_F^2\\\overset{c}{\leq}&\frac{\mu\theta_k}{2}\|\mathbf{y}^k-\mathbf{x}^*\|_F^2+\frac{(L\theta_k-\mu)\theta_k}{2}\left\|\frac{L-\mu}{L\theta_k-\mu}\mathbf{y}^k-\frac{(1-\theta_k)L}{L\theta_k-\mu}\mathbf{x}^k-\mathbf{x}^*\right\|_F^2\\\overset{d}{=}&\frac{\mu\theta_k}{2}\|\mathbf{y}^k-\mathbf{x}^*\|_F^2+\frac{(L\theta_k-\mu)\theta_k}{2}\left\|\mathbf{w}^k-\mathbf{x}^*\right\|_F^2,\end{aligned}$$

where we let $\frac{\mu}{L\theta_k}\leq 1$, and use Jensen's inequality for $\|\cdot\|_F^2$ in $\overset{c}{\leq}$, and the first identify in Lemma 8 in $\overset{d}{=}$. Plugging it into the above inequality and using the bounds for $\|\delta^k\|_F$ and $\|\Pi\delta^k\|_F$ in Lemma 7, we get (21). □

Due to the term $\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F$ on the right hand side of (21), recursion (21) cannot be directly telescoped unless we assume the boundness of $\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F$. Lemma 5 can be used to avoid such boundness assumption. Now, we use Lemmas 9 and 5 to analyze procedure (7a)-(7c). The following theorem shows the convergence for strongly convex problems.

Theorem 4: Assume that Assumptions 1, 3 and (14) hold with $\mu > 0$, and $\varepsilon_k \leq (1-(1+\tau)\theta)^{k+1}$ holds for all $k \leq K$, where $1 > \tau > 0$ can be any small constant. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\theta_k = \theta = \sqrt{\frac{\mu}{L}}$ for all k, and $\vartheta_k = (1-\theta)^{k+1}$. Then, we have

$$
\begin{aligned}
&f(\mathbf{x}^{K+1}) - f(\mathbf{x}^*) \leq C_2(1-\theta)^{K+1},\\
&\|\Pi\mathbf{x}^{K+1}\|_F \leq C_3(1-\theta)^{K+1},\\
&\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F^2 \leq C_4(1-\theta)^{K+1},\\
&f\left(\alpha(\mathbf{x}^{K+1})\right) - f(\mathbf{x}^*) \leq C_5(1-\theta)^{K+1} + \frac{LC_3^2}{2}(1-\theta)^{2K+2},
\end{aligned}
$$

where $C_2 = C_6 + \|\lambda^*\|_F C_3$, $C_3 = \frac{\sqrt{2\beta_0 C_6} + \|\lambda^*\|_F}{\beta_0}$, $C_4 = \frac{2C_6}{\mu}$, $C_5 = (\|\nabla f(\mathbf{x}^*)\|_F + L\sqrt{C_4})C_3 + C_2$ and $C_6 = \frac{18}{\tau^2\theta^2} + 2\left(f(\mathbf{x}^0) - f(\mathbf{x}^*) + \left\langle\lambda^*, \Pi\mathbf{x}^0\right\rangle\right) + \frac{1}{\beta_0}\|\beta_0\Pi\mathbf{x}^0 - \lambda^*\|_F^2 + \mu\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2$.

Proof 5: The setting of $\theta = \sqrt{\frac{\mu}{L}}$ satisfies

$$
(L\theta - \mu)\theta = L\theta^2(1-\theta). \tag{23}
$$

Sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy the requirement in Lemma 9. Define the Lyapunov function ℓ_{k+1} as follows:

$$
\ell_{k+1}^2 = \frac{\rho_{k+1} + \frac{\vartheta_k}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1} - \lambda^*\right\|_F^2 + \frac{L\theta^2}{2}\|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F^2}{(1-\theta)^{k+1}},
$$

where ρ_k is defined in (20). Dividing both sides of (21) by $(1-\theta)^{k+1}$, and using (23) and $\vartheta_k = (1-\theta)\vartheta_{k-1}$, we have

$$
\ell_{k+1}^2 - \ell_k^2 \leq \frac{\varepsilon_k}{(1-\theta)^{k+1}} + \frac{L\theta}{(1-\theta)^{k+1}}\sqrt{\frac{2\varepsilon_k}{L}}\|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F.
$$

Summing over $k = 0, 1, \cdots, K$, we have

$$
\begin{aligned}
&\ell_{K+1}^2 - \ell_0^2\\
\leq&\sum_{k=0}^{K}\frac{\varepsilon_k}{(1-\theta)^{k+1}} + \sum_{k=0}^{K}\frac{L\theta}{(1-\theta)^{k+1}}\sqrt{\frac{2\varepsilon_k}{L}}\|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F\\
=&\sum_{k=0}^{K}\frac{\varepsilon_k}{(1-\theta)^{k+1}} + \sum_{k=1}^{K+1}\frac{2\sqrt{\varepsilon_{k-1}}}{(1-\theta)^{k/2}}\sqrt{\frac{L\theta^2}{2(1-\theta)^k}}\|\mathbf{w}^k - \mathbf{x}^*\|_F\\
\overset{a}{\leq}&\sum_{k=0}^{K}\frac{\varepsilon_k}{(1-\theta)^{k+1}} + \sum_{k=1}^{K+1}\frac{2\sqrt{\varepsilon_{k-1}}}{(1-\theta)^{k/2}}\ell_k,
\end{aligned}
$$

where we use the definition of ℓ_k and $\rho_k \geq 0$ in $\overset{a}{\leq}$. Letting $s_{k+1} = \sum_{t=0}^{k}\frac{\varepsilon_t}{(1-\theta)^{t+1}} + \ell_0^2$ and $\alpha_k = \frac{2\sqrt{\varepsilon_{k-1}}}{(1-\theta)^{k/2}}$, then we have $\ell_{k+1}^2 \leq s_{k+1} + \sum_{i=1}^{k+1}\alpha_i\ell_i$ and $\ell_0^2 = s_0$. From Lemma 5, we have $\ell_{k+1} \leq \frac{1}{2}\sum_{i=1}^{k+1}\alpha_i + \sqrt{\left(\frac{1}{2}\sum_{i=1}^{k+1}\alpha_i\right)^2 + s_{k+1}}$. Letting $\varepsilon_k \leq (1-(1+\tau)\theta)^{k+1}$, and after some simple computing, we obtain

$$
\begin{aligned}
\ell_{K+1}^2 \leq&\left(\sum_{k=1}^{K+1}\frac{2\sqrt{\varepsilon_{k-1}}}{(1-\theta)^{k/2}}\right)^2 + \sum_{k=0}^{K}\frac{2\varepsilon_k}{(1-\theta)^{k+1}} + 2\ell_0^2\\
\leq&\frac{18}{\tau^2\theta^2} + 2\ell_0^2 \equiv C_6.
\end{aligned}
$$

From the definition of ℓ_{K+1} and $\rho_k \geq 0$, we get the second conclusion. From the definition of ρ_{k+1}, we have $f(\mathbf{x}^{k+1}) - f(\mathbf{x}^*) \leq \rho_{k+1} + \|\lambda^*\|_F\|\Pi\mathbf{x}^{k+1}\|_F$, which further leads to the first conclusion. Since $f(\mathbf{x}) + \langle\lambda^*, \Pi\mathbf{x}\rangle$ is μ-strongly convex over $\mathbf{x}$ and $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}} f(\mathbf{x}) + \langle\lambda^*, \Pi\mathbf{x}\rangle$, we have $\frac{\mu}{2}\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F^2 \leq f(\mathbf{x}^{K+1}) + \left\langle\lambda^*, \Pi\mathbf{x}^{K+1}\right\rangle - f(\mathbf{x}^*) - \langle\lambda^*, \Pi\mathbf{x}^*\rangle = \rho_{K+1} \leq C_6(1-\theta)^{K+1}$, i.e., the third conclusion. For the fourth conclusion, we have

$$
\begin{aligned}
&f\left(\alpha(\mathbf{x}^{K+1})\right) - f(\mathbf{x}^*)\\
=&f\left(\alpha(\mathbf{x}^{K+1})\right) - f(\mathbf{x}^{K+1}) + f(\mathbf{x}^{K+1}) - f(\mathbf{x}^*)\\
\overset{b}{\leq}&\left\langle\nabla f(\mathbf{x}^{K+1}), -\Pi\mathbf{x}^{K+1}\right\rangle + \frac{L}{2}\left\|\Pi\mathbf{x}^{K+1}\right\|_F^2 + f(\mathbf{x}^{K+1}) - f(\mathbf{x}^*)\\
\overset{c}{\leq}&\left(\|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F\right)\|\Pi\mathbf{x}^{K+1}\|_F + \frac{L}{2}\|\Pi\mathbf{x}^{K+1}\|_F^2\\
&+ f(\mathbf{x}^{K+1}) - f(\mathbf{x}^*),
\end{aligned}
\tag{24}
$$

where we use the smoothness of $f(\mathbf{x})$ and the definition of Π in $\overset{b}{\leq}$ and $\overset{c}{\leq}$. □

In the following theorem, we consider the case that $f(\mathbf{x})$ is nonstrongly convex.

Theorem 5: Assume that Assumptions 1, 3 and (14) hold with $\mu = 0$ and $\varepsilon_k \leq \frac{1}{(k+1)^6}$ for all $k \leq K$. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\theta_0 = 1$, $\frac{1-\theta_k}{\theta_k^2} = \frac{1}{\theta_{k-1}^2}$, and $\vartheta_k = \theta_k^2$. Then, we have

$$
\begin{aligned}
&f(\mathbf{x}^{K+1}) - f(\mathbf{x}^*) \leq \frac{C_7}{(K+2)^2},\\
&\|\Pi\mathbf{x}^{K+1}\|_F \leq \frac{C_8}{(K+2)^2},\\
&\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F^2 \leq C_9,\\
&f\left(\alpha(\mathbf{x}^{K+1})\right) - f(\mathbf{x}^*) \leq \frac{C_{10}}{(K+2)^2} + \frac{LC_8^2}{2(K+2)^4},
\end{aligned}
$$

where $C_7 = 4C_{11} + \|\nabla f(\mathbf{x}^*)\|_F C_8$, $C_8 = \frac{4\sqrt{2\beta_0 C_{11}} + 4\|\nabla f(\mathbf{x}^*)\|_F}{\beta_0}$, $C_9 = \frac{2C_{11}}{L}$, $C_{10} = (\|\nabla f(\mathbf{x}^*)\|_F + L\sqrt{C_9})C_8 + C_7$, and $C_{11} = 5 + \frac{\|\nabla f(\mathbf{x}^*)\|_F^2}{\beta_0} + L\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2$.

Proof 6: Define the following Lyapunov function ℓ_{k+1}

$$
\ell_{k+1}^2 = \frac{\rho_{k+1}}{\theta_k^2} + \frac{1}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}\Pi\mathbf{x}^{k+1} - \lambda^*\right\|_F^2 + \frac{L}{2}\|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F^2.
$$

Dividing both sides of (21) by θ_k^2, using $\vartheta_k = \theta_k^2$ and $\frac{1-\theta_k}{\theta_k^2} = \frac{1}{\theta_{k-1}^2}$, we have

$$
\ell_{k+1}^2 - \ell_k^2 \leq \frac{\varepsilon_k}{\theta_k^2} + \frac{L}{\theta_k}\sqrt{\frac{2\varepsilon_k}{L}}\|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F.
$$

Similar to the proof of Theorem 4, we obtain

$$
\ell_{K+1}^2 - \ell_0^2 \leq \sum_{k=0}^{K}\frac{\varepsilon_k}{\theta_k^2} + \sum_{k=1}^{K+1}\frac{2\sqrt{\varepsilon_{k-1}}}{\theta_{k-1}}\ell_k.
$$

From Lemma 5 and a similar induction to Theorem 4, we have

$$
\begin{aligned}
&\ell_{K+1}^2\\
\leq&\left(\sum_{k=1}^{K+1}\frac{2\sqrt{\varepsilon_{k-1}}}{\theta_{k-1}}\right)^2 + \sum_{k=0}^{K}\frac{2\varepsilon_k}{\theta_k^2} + 2\ell_0^2\\
\overset{a}{\leq}&\left(\sum_{k=1}^{K+1}2k\sqrt{\varepsilon_{k-1}}\right)^2 + \sum_{k=0}^{K}2\varepsilon_k(k+1)^2 + \frac{\|\lambda^*\|_F^2}{\beta_0} + L\|\mathbf{w}^0 - \mathbf{x}^*\|_F^2,
\end{aligned}
$$

where we use $\frac{1}{k+1} \leq \theta_k \leq \frac{2}{k+2}$ and $\frac{1}{\theta_{-1}^2} = 0$ in $\overset{a}{\leq}$, which can be derived from $\frac{1-\theta_k}{\theta_k^2} = \frac{1}{\theta_{k-1}^2}$ and $\theta_0 = 1$. Letting $\varepsilon_k \leq \frac{1}{(k+1)^{4+2\tau}}$, then we have $\sum_{k=0}^{K} 2\varepsilon_k (k+1)^2 \leq \frac{2}{1+2\tau}$ and $\sum_{k=1}^{K+1} 2k\sqrt{\varepsilon_{k-1}} \leq \frac{2}{\tau}$. So

$$\ell_{K+1}^2 \leq \frac{4}{\tau^2} + \frac{4}{1+2\tau} + \frac{\|\lambda^*\|_F^2}{\beta_0} + L\|\mathbf{x}^0 - \mathbf{x}^*\|_F^2 \equiv C_{11},$$

where we let $\tau = 1$ for simplicity and use Lemma 2. From the definition of $\mathbf{w}^{k+1} = \frac{\mathbf{x}^{k+1}}{\theta_k} - \frac{1-\theta_k}{\theta_k}\mathbf{x}^k$, we have $\|\mathbf{x}^{k+1} - \mathbf{x}^*\|_F = \|\theta_k \mathbf{w}^{k+1} + (1-\theta_k)\mathbf{x}^k - \mathbf{x}^*\|_F \leq \theta_k \|\mathbf{w}^{k+1} - \mathbf{x}^*\|_F + (1-\theta_k)\|\mathbf{x}^k - \mathbf{x}^*\|_F$. By induction, we can prove $\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F^2 \leq \frac{2C_{11}}{L}$ for any k. Similar to the proof of Theorem 4 and using Lemma 2, we have the remaining conclusions. □

3) Total Numbers of Communications and Computations: Based on Theorems 4 and 5, and the inner loop iteration number given in (18), we can establish the gradient computation and communication complexities for Algorithm 1. We first consider the strongly convex case and prove Theorem 1.

Proof 7: $\|\Pi \mathbf{z}^k\|_F$ appears in (18). We first prove that $\|\Pi \mathbf{z}^k\|_F$ is bounded for any k given $T_k = \frac{1}{-2\log\left(1-\sqrt{1-\sigma_2(W)}\right)} \log\left(\frac{\beta_0}{2\vartheta_k \varepsilon_k}\left(\frac{1}{L}\|\nabla f(\mathbf{x}^*)\|_F + 6\sqrt{C_4}\right)^2\right)$, where C_4 is defined in Theorem 4. We prove $\|\Pi \mathbf{z}^k\|_F \leq \frac{1}{L}\|\nabla f(\mathbf{x}^*)\|_F + 6\sqrt{C_4}$ by induction. The case for $k = 0$ can be easily verified since $\|\Pi \mathbf{z}^0\|_F = \|\Pi \mathbf{x}^0 - \Pi \mathbf{x}^*\|_F \leq \|\mathbf{x}^0 - \mathbf{x}^*\|_F$. Assume that the conclusion holds for all $k \leq K$. Then from (18) we know that (14) holds for $k \leq K$. From Theorem 4, we have $\|\mathbf{x}^K - \mathbf{x}^*\|_F \leq \sqrt{C_4}$ and $\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F \leq \sqrt{C_4}$. Thus,

$$\begin{aligned}
&\|\Pi \mathbf{z}^{K+1}\|_F \\
\overset{a}{\leq} &\|\Pi \mathbf{y}^{K+1}\|_F + \frac{1}{L}\|\nabla f(\mathbf{y}^{K+1})\|_F \\
\overset{b}{\leq} &\|\Pi(\mathbf{y}^{K+1} - \mathbf{x}^*)\|_F + \frac{1}{L}\left(\|\nabla f(\mathbf{x}^*)\|_F + L\|\mathbf{y}^{K+1} - \mathbf{x}^*\|_F\right) \\
\leq &\frac{1}{L}\|\nabla f(\mathbf{x}^*)\|_F + 2\|\mathbf{y}^{K+1} - \mathbf{x}^*\|_F \\
\overset{c}{\leq} &\frac{1}{L}\|\nabla f(\mathbf{x}^*)\|_F + 4\|\mathbf{x}^{K+1} - \mathbf{x}^*\|_F + 2\|\mathbf{x}^K - \mathbf{x}^*\|_F \\
\leq &\frac{1}{L}\|\nabla f(\mathbf{x}^*)\|_F + 6\sqrt{C_4},
\end{aligned}$$

where we use (7b) in $\overset{a}{\leq}$, the smoothness of $f(\mathbf{x})$ and $\Pi \mathbf{x}^* = 0$ in $\overset{b}{\leq}$, and $\mathbf{y}^k = \mathbf{x}^k + \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}(\mathbf{x}^k - \mathbf{x}^{k-1})$ in $\overset{c}{\leq}$, which is equivalent to (7a) with the special setting of θ_k. So we get the conclusion.

From Theorem 4, to find a solution satisfying (9), we know that the number of gradient computations, i.e., the number of outer iterations, is $O\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$. From (18), we have

$$\begin{aligned}
T_k =& O\left(\frac{1}{-\log\left(1-\sqrt{1-\sigma_2(W)}\right)}\log\frac{1}{(1-\theta)^{2(k+1)}}\right) \\
=& O\left(\frac{k\log\left(1-\sqrt{\mu/L}\right)}{\log\left(1-\sqrt{1-\sigma_2(W)}\right)}\right) \overset{c}{=} O\left(\frac{k\sqrt{\mu/L}}{\sqrt{1-\sigma_2(W)}}\right),
\end{aligned}$$

where we use $\log\left((1-\sqrt{1-\sigma_2(W)})\right) \approx -\sqrt{1-\sigma_2(W)}$ and $\log\left(1-\sqrt{\mu/L}\right) \approx -\sqrt{\mu/L}$ in $\overset{c}{=}$ from Taylor expansion when $\sqrt{1-\sigma_2(W)}$ and $\sqrt{\mu/L}$ are small. Thus, the total number of communications, i.e., the total number of inner iterations, is

$$\sum_{k=0}^{\sqrt{L/\mu}\log\frac{1}{\epsilon}} O\left(k\sqrt{\frac{\mu}{L(1-\sigma_2(W))}}\right) = O\left(\sqrt{\frac{L}{\mu(1-\sigma_2(W))}}\log^2\frac{1}{\epsilon}\right).$$

The proof is complete. □

Similar to the proof of Theorem 1, we can also prove Theorem 2 for the nonstrongly case.

Proof 8: Similar to the above proof of Theorem 1 and given the similar T_k replacing C_4 by C_9, we know that $\|\Pi \mathbf{z}^k\|_F$ is also bounded for all k. Let $\beta_0 \geq L + L\|\nabla f(\mathbf{x}^*)\|_F^2$, and assume $L \geq 1$ and $\|\nabla f(\mathbf{x}^*)\|_F \geq 1$ for simplicity. Using the constants in (13), we know $C_7 = O(mLR_1^2)$, $C_8 = O(\sqrt{m}R_1)$, $C_9 = O(mR_1^2)$, and $C_{10} = O(mLR_1^2)$. Let $\epsilon = \frac{LR_1^2}{(K+2)^2}$. From Theorem 5, we know that Algorithm 1 needs $O\left(\sqrt{\frac{L}{\epsilon}}\right)$ gradient computations such that $\frac{1}{m}\left(f\left(\alpha(\mathbf{x}^{K+1})\right) - f(\mathbf{x}^*)\right) \leq \epsilon$ and $\frac{1}{m}\|\Pi \mathbf{x}^{K+1}\|_F^2 \leq \epsilon^2$, i.e., (9) holds. From (18), we have

$$T_k = O\left(\frac{\log(k+1)^8}{-\log\left(1-\sqrt{1-\sigma_2(W)}\right)}\right) = O\left(\frac{\log k}{\sqrt{1-\sigma_2(W)}}\right),$$

Thus, the total number of communications is

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

The proof is complete. □

C. Complexity Analysis for Algorithm 2

Now we prove Theorem 3. Similar to Section III-B, we define

$$\rho_{k+1} = F(\mathbf{x}^{k+1}) - F(\mathbf{x}^*) + \left\langle \lambda^*, U\mathbf{x}^{k+1}\right\rangle,$$

where $(\mathbf{x}^*, \lambda^*)$ is a pair of KKT points of saddle point problem $\min_{\mathbf{x}}\max_{\lambda} F(\mathbf{x}) + \langle \lambda, U\mathbf{x}\rangle$ satisfying Lemma 2. Define

$$\mathbf{w}^{k+1} \equiv \frac{\mathbf{x}^{k+1}}{\theta_k} - \frac{1-\theta_k}{\theta_k}\mathbf{x}^k \text{ for any } k \geq 0 \text{ and } \mathbf{w}^0 = \mathbf{x}^0.$$

From the definitions of $\mathbf{w}^{k+1}$ and $\mathbf{y}^k$ in (11a), we have the following easy-to-identify identities.

Lemma 10: For procedure (11a)-(11c), we have

$$\begin{aligned}
&\theta_k \mathbf{x}^* + (1-\theta_k)\mathbf{x}^k - \mathbf{y}^k = \theta_k\left(\mathbf{x}^* - \mathbf{w}^k\right), \\
&\theta_k \mathbf{x}^* + (1-\theta_k)\mathbf{x}^k - \mathbf{x}^{k+1} = \theta_k\left(\mathbf{x}^* - \mathbf{w}^{k+1}\right).
\end{aligned} \tag{25}$$

We use the same notations of ρ_{k+1} and $\mathbf{w}^k$ with Section III-B for easy analogy. Different from Section III-B, we define a new variable

$$\mathbf{v}^{k,t} \equiv \frac{\mathbf{z}^{k,t}}{\theta_k} - \frac{1-\theta_k}{\theta_k}\mathbf{x}^k. \tag{26}$$

The proof of Theorem 3 is based on the following Lyapunov function

$$\ell_{k+1}=\frac{\rho_{k+1}}{\theta_k}+\frac{1}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{x}^{k+1}-\lambda^*\right\|_F^2 +\left(\frac{L\theta_{k+1}}{2}+\frac{\beta_0}{2}\right)\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F^2+\frac{M}{2\sqrt{1-\sigma_2(W)}}\|\mathbf{v}^{k+1,0}-\mathbf{x}^*\|_F^2.$$

Analogy to Lemma 9, we give the following lemma, which describes a progress in one iteration of Algorithm 2.

Lemma 11: Assume that Assumptions 1, 2 and 3 hold with $\mu=0$. Let sequences $\{\theta_k\}$ and $\{\vartheta_k\}$ satisfy $\theta_0=1$, $\frac{1-\theta_k}{\theta_k}=\frac{1}{\theta_{k-1}}$, and $\vartheta_k=\theta_k$. Assume the following equation holds

$$\frac{\theta_k}{\eta_k T_k}=\frac{M}{\sqrt{1-\sigma_2(W)}}. \tag{27}$$

Then, for Algorithm 2, we have

$$\ell_{k+1}\le\ell_k+\frac{mM^2\eta_k}{2\theta_k}. \tag{28}$$

The proof of Lemma 11 is based on the following lemma.

Lemma 12: Assume that Assumptions 1 and 3 hold. Define $\widetilde{\mathbf{x}}^{k,*}=(1-\theta_k)\mathbf{x}^k+\theta_k\mathbf{x}^*$. Then, for Algorithm 2, we have

$$\begin{aligned}&\rho_{k+1}-(1-\theta_k)\rho_k\\ &\le\left\langle\nabla f(\mathbf{y}^k),\mathbf{x}^{k+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2\\ &\quad+h(\mathbf{x}^{k+1})-h(\widetilde{\mathbf{x}}^{k,*})+\left\langle\lambda^*,U\mathbf{x}^{k+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle.\end{aligned}$$

Proof 9: From (22) with $\mu=0$, we have

$$f(\mathbf{x}^{k+1})\le f(\mathbf{x})+\left\langle\nabla f(\mathbf{y}^k),\mathbf{x}^{k+1}-\mathbf{x}\right\rangle+\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2.$$

Firstly let $\mathbf{x}=\mathbf{x}^k$ and then $\mathbf{x}=\mathbf{x}^*$, we obtain two inequalities. Multiplying the first inequality by $(1-\theta_k)$, multiplying the second by θ_k, and adding them together, we have

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

Adding $h(\mathbf{x}^{k+1})-(1-\theta_k)h(\mathbf{x}^k)-\theta_k h(\mathbf{x}^*)+\left\langle\lambda^*,U\mathbf{x}^{k+1}-(1-\theta_k)U\mathbf{x}^k\right\rangle$ to both sides, and using the definition of ρ_k, we have

$$\begin{aligned}&\rho_{k+1}-(1-\theta_k)\rho_k\\ &=F(\mathbf{x}^{k+1})-(1-\theta_k)F(\mathbf{x}^k)-\theta_k F(\mathbf{x}^*)\\ &\quad+\left\langle\lambda^*,U\mathbf{x}^{k+1}-(1-\theta_k)U\mathbf{x}^k\right\rangle\\ &\le\left\langle\nabla f(\mathbf{y}^k),\mathbf{x}^{k+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\right\rangle+\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2\\ &\quad+h(\mathbf{x}^{k+1})-(1-\theta_k)h(\mathbf{x}^k)-\theta_k h(\mathbf{x}^*)\\ &\quad+\left\langle\lambda^*,U\mathbf{x}^{k+1}-(1-\theta_k)U\mathbf{x}^k\right\rangle.\end{aligned} \tag{29}$$

From the definition of $\widetilde{\mathbf{x}}^{k,*}$, $U\mathbf{x}^*=0$, and the convexity of $h(\mathbf{x})$, we have

$$\begin{aligned}&\mathbf{x}^{k+1}-\widetilde{\mathbf{x}}^{k,*}=\mathbf{x}^{k+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*,\\ &U\mathbf{x}^{k+1}-U\widetilde{\mathbf{x}}^{k,*}=U\mathbf{x}^{k+1}-(1-\theta_k)U\mathbf{x}^k,\\ &h(\widetilde{\mathbf{x}}^{k,*})\le(1-\theta_k)h(\mathbf{x}^k)+\theta_k h(\mathbf{x}^*).\end{aligned}$$

Plugging them into (29), we have the conclusion. □

Now, we give the proof of Lemma 11.

Proof 10: From the fact that $h(\mathbf{x})$ is $(\sqrt{m}M)$-Lipchitz continuous derived by Assumption 2, similar to the induction in (22), we have

$$\begin{aligned}&h(\mathbf{z}^{k,t+1})\\ &\le h(\mathbf{z}^{k,t})+\left\langle\hat{\nabla}h(\mathbf{z}^{k,t}),\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\right\rangle+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F\\ &=h(\mathbf{z}^{k,t})+\left\langle\hat{\nabla}h(\mathbf{z}^{k,t}),\widetilde{\mathbf{x}}^{k,*}-\mathbf{z}^{k,t}\right\rangle\\ &\quad+\left\langle\hat{\nabla}h(\mathbf{z}^{k,t}),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F\\ &\le h(\widetilde{\mathbf{x}}^{k,*})+\left\langle\hat{\nabla}h(\mathbf{z}^{k,t}),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F,\end{aligned} \tag{30}$$

where $\widetilde{\mathbf{x}}^{k,*}$ is defined in Lemma 12 and $\hat{\nabla}h(\mathbf{z}^{k,t})\in\partial h(\mathbf{z}^{k,t})$. On the other hand, from the update role of $\mathbf{z}^{k,t+1}$ in Algorithm 2, we have

$$\begin{aligned}0=&\hat{\nabla}h(\mathbf{z}^{k,t})+\nabla f(\mathbf{y}^k)+\frac{\beta_0}{\vartheta_k}U^2\mathbf{y}^k\\ &+\left(L+\frac{\beta_0}{\vartheta_k}\right)(\mathbf{z}^{k,t+1}-\mathbf{y}^k)+\frac{1}{\eta_k}(\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}).\end{aligned} \tag{31}$$

Thus, we have

$$\begin{aligned}&\left\langle\nabla f(\mathbf{y}^k),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\frac{L}{2}\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2\\ &+h(\mathbf{z}^{k,t+1})-h(\widetilde{\mathbf{x}}^{k,*})+\left\langle\lambda^*,U\mathbf{z}^{k,t+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\overset{a}{\le}\left\langle\nabla f(\mathbf{y}^k)+\hat{\nabla}h(\mathbf{z}^{k,t}),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\left\langle\lambda^*,U\mathbf{z}^{k,t+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\quad+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F+\frac{L}{2}\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2\\ &\overset{b}{=}-\left\langle\frac{\beta_0}{\vartheta_k}U^2\mathbf{y}^k+\left(L+\frac{\beta_0}{\vartheta_k}\right)(\mathbf{z}^{k,t+1}-\mathbf{y}^k)\right.\\ &\qquad\left.+\frac{1}{\eta_k}(\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\quad+\left\langle\lambda^*,U\mathbf{z}^{k,t+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F\\ &\quad+\frac{L}{2}\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2\\ &=-\left\langle\frac{\beta_0}{\vartheta_k}U\mathbf{y}^k-\lambda^*,U\mathbf{z}^{k,t+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\quad-\left(L+\frac{\beta_0}{\vartheta_k}\right)\left\langle\mathbf{z}^{k,t+1}-\mathbf{y}^k,\mathbf{y}^k-\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\quad-\frac{1}{\eta_k}\left\langle\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t},\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle\\ &\quad+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F-\left(\frac{L}{2}+\frac{\beta_0}{\vartheta_k}\right)\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2\\ &\overset{c}{=}-\frac{\vartheta_k}{\beta_0}\left\langle\frac{\beta_0}{\vartheta_k}U\mathbf{y}^k-\lambda^*,\frac{\beta_0}{\vartheta_k}U\mathbf{z}^{k,t+1}-\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k\right\rangle\\ &\quad-\left(L+\frac{\beta_0}{\vartheta_k}\right)\left\langle\mathbf{z}^{k,t+1}-\mathbf{y}^k,\mathbf{y}^k-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\right\rangle\\ &\quad-\frac{1}{\eta_k}\left\langle\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t},\mathbf{z}^{k,t+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\right\rangle\\ &\quad+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F-\left(\frac{L}{2}+\frac{\beta_0}{\vartheta_k}\right)\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2,\end{aligned}$$

where we use (30) in $\overset{a}{\le}$, (31) in $\overset{b}{=}$, $\frac{1}{\vartheta_{k-1}}=\frac{1-\theta_k}{\vartheta_k}$, and the definition of $\widetilde{\mathbf{x}}^{k,*}$ in $\overset{c}{=}$. Applying the identities in Lemma 1 to the two inner products, using

$\frac{\vartheta_k}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{y}^k-\frac{\beta_0}{\vartheta_k}U\mathbf{z}^{k,t+1}\right\|_F^2 \le \frac{\beta_0}{2\vartheta_k}\|\mathbf{y}^k-\mathbf{z}^{k,t+1}\|_F^2$ and dropping the negative term $-\frac{\vartheta_k}{2\beta_0}\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{y}^k-\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k\right\|_F^2$, we have

$$
\begin{aligned}
&\left\langle\nabla f(\mathbf{y}^k),\mathbf{z}^{k,t+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\frac{L}{2}\|\mathbf{z}^{k,t+1}-\mathbf{y}^k\|_F^2\\
&+h(\mathbf{z}^{k,t+1})-h(\widetilde{\mathbf{x}}^{k,*})+\left\langle\lambda^*,U\mathbf{z}^{k,t+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle\\
\le&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{z}^{k,t+1}-\lambda^*\right\|_F^2\right]\\
&+\left(\frac{L}{2}+\frac{\beta_0}{2\vartheta_k}\right)\left[\|\mathbf{y}^k-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{z}^{k,t+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]\\
&+\frac{1}{2\eta_k}\left[\|\mathbf{z}^{k,t}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{z}^{k,t+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]\\
&+\sqrt{m}M\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F-\frac{1}{2\eta_k}\|\mathbf{z}^{k,t+1}-\mathbf{z}^{k,t}\|_F^2\\
\overset{d}{\le}&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{z}^{k,t+1}-\lambda^*\right\|_F^2\right]\\
&+\left(\frac{L}{2}+\frac{\beta_0}{2\vartheta_k}\right)\left[\|\mathbf{y}^k-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{z}^{k,t+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]\\
&+\frac{1}{2\eta_k}\left[\|\mathbf{z}^{k,t}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{z}^{k,t+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]+\frac{mM^2\eta_k}{2},
\end{aligned}
$$

where we use $-\frac{a}{2}t^2+bt\le\frac{b^2}{2a}$ for any $a>0$ in $\overset{d}{\le}$. Summing over $t=0,\cdots,T_k-1$ and dividing both sides by T_k, letting $\mathbf{x}^{k+1}=\frac{\sum_{t=0}^{T_k-1}\mathbf{z}^{k,t+1}}{T_k}$, and from the convexity of $h(\mathbf{x})$ and $\|\cdot\|_F^2$, we have

$$
\begin{aligned}
&\left\langle\nabla f(\mathbf{y}^k),\mathbf{x}^{k+1}-\widetilde{\mathbf{x}}^{k,*}\right\rangle+\frac{L}{2}\|\mathbf{x}^{k+1}-\mathbf{y}^k\|_F^2\\
&+h(\mathbf{x}^{k+1})-h(\widetilde{\mathbf{x}}^{k,*})+\left\langle\lambda^*,U\mathbf{x}^{k+1}-U\widetilde{\mathbf{x}}^{k,*}\right\rangle\\
\le&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{x}^{k+1}-\lambda^*\right\|_F^2\right]\\
&+\left(\frac{L}{2}+\frac{\beta_0}{2\vartheta_k}\right)\left[\|\mathbf{y}^k-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{x}^{k+1}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]\\
&+\frac{1}{2\eta_kT_k}\left[\|\mathbf{z}^{k,0}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right.\\
&\qquad\left.-\|\mathbf{z}^{k,T_k}-(1-\theta_k)\mathbf{x}^k-\theta_k\mathbf{x}^*\|_F^2\right]+\frac{mM^2\eta_k}{2}\\
\overset{e}{=}&\frac{\vartheta_k}{2\beta_0}\left[\left\|\frac{\beta_0}{\vartheta_{k-1}}U\mathbf{x}^k-\lambda^*\right\|_F^2-\left\|\frac{\beta_0}{\vartheta_k}U\mathbf{x}^{k+1}-\lambda^*\right\|_F^2\right]\\
&+\left(\frac{L}{2}+\frac{\beta_0}{2\vartheta_k}\right)\theta_k^2\left[\|\mathbf{w}^k-\mathbf{x}^*\|_F^2-\|\mathbf{w}^{k+1}-\mathbf{x}^*\|_F^2\right]\\
&+\frac{\theta_k^2}{2\eta_kT_k}\left[\|\mathbf{v}^{k,0}-\mathbf{x}^*\|_F^2-\|\mathbf{v}^{k+1,0}-\mathbf{x}^*\|_F^2\right]+\frac{mM^2\eta_k}{2},
\end{aligned}
$$

where $\overset{e}{=}$ follows from the identities in Lemma 25, the definition of $\mathbf{v}^{k,t}$ in (26), and $\mathbf{z}^{k+1,0}=\mathbf{z}^{k,T_k}$. Dividing both sides by θ_k and letting $\vartheta_k=\theta_k$, from Lemma 12, $\frac{1}{\theta_{k-1}}=\frac{1-\theta_k}{\theta_k}$, $\frac{\theta_k}{2\eta_kT_k}=\frac{M}{2\sqrt{1-\sigma_2(W)}}$, $\theta_{k+1}\le\theta_k$, and the definition of ℓ_k, we have the conclusion. □

Based on Lemma 11, we can prove Theorem 3.

Proof 11: The settings of $T_k=K(1-\sigma_2(W))$ and $\eta_k=\frac{\theta_k}{KM\sqrt{1-\sigma_2(W)}}$ satisfy (27). Plugging them into (28), we have

$$\ell_{k+1}\le\ell_k+\frac{mM}{2K\sqrt{1-\sigma_2(W)}}.$$

Summing over $k=0,\cdots,K-1$, we have

$$
\begin{aligned}
\ell_K\le&\ell_0+\frac{mM}{2\sqrt{1-\sigma_2(W)}}\\
=&\frac{1}{2\beta_0}\|\lambda^*\|_F^2+\frac{L+\beta_0}{2}\|\mathbf{x}^0-\mathbf{x}^*\|_F^2\\
&+\frac{M}{2\sqrt{1-\sigma_2(W)}}\|\mathbf{x}^0-\mathbf{x}^*\|_F^2+\frac{mM}{2\sqrt{1-\sigma_2(W)}}\\
\equiv&C_{12},
\end{aligned}
$$

where we use $\theta_0=1$, $\frac{1}{\theta_{-1}}=\frac{1-\theta_0}{\theta_0}=0$, $\mathbf{w}^0=\mathbf{x}^0$, and $\mathbf{v}^{0,0}=\mathbf{x}^0$. Similar to the proofs of Theorems 4 and 5, from the definition of ℓ_k and $\theta_{k-1}=\frac{1}{k}$, we have

$$
\begin{aligned}
&\|U\mathbf{x}^K\|_F\le\frac{1}{\beta_0K}\left(\sqrt{2\beta_0C_{12}}+\|\lambda^*\|_F\right),\\
&F(\mathbf{x}^K)-F(\mathbf{x}^*)\le\frac{C_{12}}{K}+\|\lambda^*\|_F\|U\mathbf{x}^K\|_F
\end{aligned}
$$

and $\|\mathbf{x}^K-\mathbf{x}^*\|_F^2\le\frac{2C_{12}}{\beta_0}$. Similar to (24), we also have

$$
\begin{aligned}
&F\left(\alpha(\mathbf{x}^K)\right)-F(\mathbf{x}^*)\\
\overset{a}{\le}&\left(\|\nabla f(\mathbf{x}^*)\|+L\sqrt{\frac{2C_{12}}{\beta_0}}\right)\|\Pi\mathbf{x}^K\|_F+\frac{L}{2}\|\Pi\mathbf{x}^K\|_F^2\\
&+2\sqrt{m}M\|\Pi\mathbf{x}^K\|_F+F(\mathbf{x}^K)-F(\mathbf{x}^*),
\end{aligned}
$$

where we use the fact that $h(\mathbf{x})$ is $(\sqrt{m}M)$-Lipchitz continuous in $\overset{a}{\le}$, i.e, $\|\widetilde{\nabla}h(\mathbf{x})\|_F\le\sqrt{m}M,\forall\widetilde{\nabla}h(\mathbf{x})\in\partial h(\mathbf{x})$. From Lemma 4, we can further bound $\|\Pi\mathbf{x}^K\|_F$ by $\frac{\|U\mathbf{x}^K\|_F}{\sqrt{1-\sigma_2(W)}}$. From Lemma 2, we know $\|\lambda^*\|_F\le\frac{\sqrt{m}M+\|\nabla f(\mathbf{x}^*)\|_F}{\sqrt{1-\sigma_2(W)}}\equiv\frac{1}{\chi}$. From the setting of β_0, we have $\beta_0\ge\frac{L}{\sqrt{1-\sigma_2(W)}}\ge L$ and $\beta_0\ge\frac{M}{\sqrt{1-\sigma_2(W)}}$. Combing with (13) and $R_1\ge1$, we have $\frac{1}{\chi}\le\sqrt{m}\beta_0\left(R_1+\frac{R_2}{L}\right)$ and

$$
\begin{aligned}
&C_{12}\le\frac{1}{2\beta_0\chi^2}+\frac{3\beta_0mR_1^2}{2}+\frac{\beta_0m}{2}\le2.5\beta_0m\left(R_1+\frac{R_2}{L}\right)^2,\\
&\|U\mathbf{x}^K\|_F\le\frac{1}{K}\left(\sqrt{5m}\left(R_1+\frac{R_2}{L}\right)+\frac{1}{\chi\beta_0}\right)\le\frac{4\sqrt{m}}{K}\left(R_1+\frac{R_2}{L}\right),\\
&\|\Pi\mathbf{x}^K\|_F\le\frac{4\beta_0\sqrt{m}}{KL}\left(R_1+\frac{R_2}{L}\right),\|\Pi\mathbf{x}^K\|_F\le\frac{4\beta_0\sqrt{m}}{KM}\left(R_1+\frac{R_2}{L}\right),\\
&F(\mathbf{x}^K)-F(\mathbf{x}^*)\le\frac{7\beta_0m}{K}\left(R_1+\frac{R_2}{L}\right)^2.
\end{aligned}
$$

Thus, we further have

$$\begin{aligned}
&F\left(\alpha(\mathbf{x}^K)\right) - F(\mathbf{x}^*)\\
\leq &\left(\sqrt{m}R_2 + L\left(R_1+\frac{R_2}{L}\right)\sqrt{5m}\right)\frac{4\beta_0\sqrt{m}}{KL}\left(R_1+\frac{R_2}{L}\right)\\
&+\frac{8\beta_0 m}{K^2\sqrt{1-\sigma_2(W)}}\left(R_1+\frac{R_2}{L}\right)^2+2\sqrt{m}\frac{4\beta_0\sqrt{m}}{K}\left(R_1+\frac{R_2}{L}\right)\\
&+\frac{7\beta_0 m}{K}\left(R_1+\frac{R_2}{L}\right)^2\\
\leq &\left(\frac{31\beta_0 m}{K}+\frac{8\beta_0 m}{K^2\sqrt{1-\sigma_2(W)}}\right)\left(R_1+\frac{R_2}{L}\right)^2.
\end{aligned}$$

The proof is complete. □

Similar to the proof of Theorem 3, in the following, we give the proof of Corollary 1.

Proof 12: The settings of $T_k = \frac{1-\sigma_2(W)}{\theta_k}$ and $\eta_k = \frac{\theta_k^2}{M\sqrt{1-\sigma_2(W)}}$ satisfy (27). Plugging them into (28) and using $\theta_k = \frac{1}{k+1}$, we have

$$\ell_{k+1} \leq \ell_k + \frac{mM}{2(k+1)\sqrt{1-\sigma_2(W)}}.$$

Summing over $k = 0, \cdots, K-1$, we have

$$\ell_K \leq \ell_0 + \frac{mM(\log K + 1)}{\sqrt{1-\sigma_2(W)}}.$$

Similar to the proof of Theorem 3, we have the conclusion. □

IV. Numerical Experiments

A. Smooth Problem

We test the performance of the proposed algorithms on the following least square regression problem

$$\min_{x\in\mathbb{R}^n}\sum_{i=1}^m f_i(x) \quad \text{with} \quad f_i(x) \equiv \frac{1}{2}\|A_i^T x - b_i\|^2 + \frac{\mu}{2}\|x\|^2. \tag{32}$$

We generate $A_i \in \mathbb{R}^{n\times N/m}$ from the uniform distribution with each entry in $[0,1]$ and normalize each column of A_i to be 1, where N is the sample size. We set $N = 1000$, $n = 500$, $m = 100$, and $b_i = A_i^T x$ with some x generated from the Gaussian distribution. We consider both the strongly convex objective ($\mu > 0$) and nonstrongly convex objective ($\mu = 0$).

We consider the Erdős−Rényi random graph where each pair of agents has a connection with the probability of p. Almost all Erdős−Rényi random graph with $p = \frac{2\log m}{m}$ is connected and $\frac{1}{1-\sigma_2(W)} = O(1)$ [34, Proposition 5]. We test the performance with $p = 0.5$, $p = 0.1$, and $p = 0.05$, and observe that $1-\sigma_2(W) = 0.33$, $1-\sigma_2(W) = 0.13$, and $1-\sigma_2(W) = 0.04$, respectively. We set $W = \frac{I+M}{2}$, where M is the Metropolis weight matrix [53].

For the strongly convex objective, we compare APM-C with the accelerated dual ascent (ADA) [12], distributed Nesterov's gradient descent (DNGD) [7], EXTRA [20], and NEAR-DGD+ [22]. NEAR-DGD+ can be seen as a counterpart of APM-C without Nesterov's acceleration scheme and accelerated average consensus. We set $\mu = 0.0001$ and leave the test on different condition numbers in our supplementary material. We set the inner iteration number T_k as $\lceil\frac{k\sqrt{\mu/L}}{3\sqrt{1-\sigma_2(W)}}\rceil$, $\beta_0 = 100$ and the stepsize as $\frac{1}{L}$ for APM-C, where $\lceil\cdot\rceil$ means the top integral function. For ADA, we follow the theory in [9] to set the inner iteration number as $\lceil\sqrt{\frac{L}{\mu}}\log\frac{L}{\mu}\rceil$ (we leave the test on the impact of smaller inner iteration numbers in our supplementary material) and the stepsize as μ. We tune the best stepsize as $\frac{1}{L}$ and $\frac{0.5}{L}$ for EXTRA and DNGD, respectively. We follow [22] to set $T_k = k$ for NEAR-DGD+. We initialize $\mathbf{x}^0$ at 0 for all the compared methods.

Figure 1 plots the comparisons. We can see that APM-C has the lowest computation cost and ADA has the lowest communication cost, which match the theory. Thus, APM-C is more suited to the environment where computation is the bottleneck of the overall performance. Due to the large T_k for ADA, it only performs several outer iterations after 3000 gradient computations and thus has almost no decreasing in the first, third and fifth plots of Figure 1. APM-C has a higher communication cost than DNGD but a lower computation cost for $p = 0.1$ and $p = 0.5$. APM-C performs better than NEAR-DGD+ and it verifies that Nesterov's acceleration scheme is critical to improve the performance. From Figure 1, we observe that APM-C is more suited to the network with small $\frac{1}{\sqrt{1-\sigma_2(W)}}$, otherwise, the communication costs will be high, e.g., see the right two plots in Figure 1. In fact, when $\frac{1}{\sqrt{1-\sigma_2(W)}}$ is small, $\frac{\sqrt{\mu/L}}{\sqrt{1-\sigma_2(W)}}$ will also be small, e.g., 0.01 in our experiment with $p = 0.1$. Thus the required T_k is small, e.g., $T_{3000} = 11$ in our experiment. As a comparison, NEAR-DGD+ suggests $T_k = k$ and thus it increases quickly, which leads to almost no decreasing in the second, fourth and sixth plots of Figure 1. In practice, we can use the expander graph [54] which satisfies $\frac{1}{1-\sigma_2(W)} = O(1)$ [34]. The Erdős−Rényi random graph is a special case of the expander graph and can be easily implemented.

For the nonstrongly convex objective, we test the performance of APM, APM-C, D-NG [8], D-NC [8], DNGD [7], EXTRA [20] and ADA [9]. We set T_k as $\lceil\frac{\log(k+1)}{5\sqrt{1-\sigma_2(W)}}\rceil$ and $\lceil\frac{\log(k+1)}{-5\log\sigma_2(W)}\rceil$ for APM-C and D-NC, respectively. We set the stepsize as $\frac{1}{L}$ for the two algorithms and $\beta_0 = 100$ for APM-C. We set $\frac{\beta_0}{\vartheta_k} = \frac{k+1}{c}$ with $c = 50$ for APM and tune the best $c = 1$ for D-NG. Larger c makes D-NG diverge. We tune the best stepsize as $\frac{1}{L}$ for EXTRA, $\frac{0.05}{L}$ for DNGD with $p = 0.05$, $\frac{0.1}{L}$ for DNGD with $p = 0.1$, and $\frac{0.2}{L}$ for DNGD with $p = 0.5$, respectively. For ADA, we follow [9] to add a small regularizer of $\frac{\epsilon}{2}\|\mathbf{x}\|^2$ to each $f_i(\mathbf{x})$ and solve a regularized problem with $\epsilon = 10^{-7}$. We set the inner iteration number as $T_k = \lceil\sqrt{\frac{L}{\epsilon}}\log\frac{L}{\epsilon}\rceil$.

From figure 2, we can see that APM-C also has the lowest computation cost. APM performs better than D-NG because APM allows to use a larger stepsize in practice, which can reduce the negative impacts from the diminishing stepsize. APM is more suited to the environment where high precision is not required, otherwise, the diminishing stepsize makes the algorithm slow. ADA has the lowest communication cost. However, ADA needs to predefine ϵ to set the algorithm parameter and thus it only achieves an approximate optimal

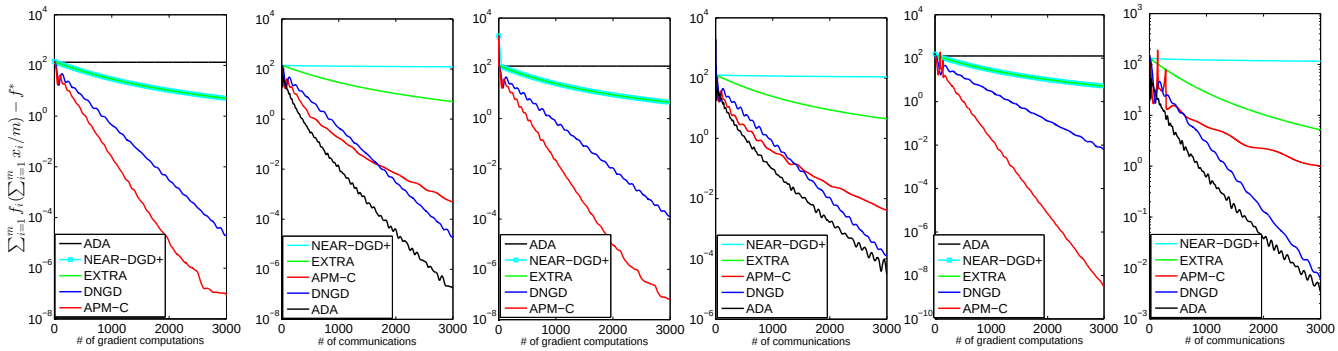


Fig. 1. Comparisons on the strongly convex problem (32) and Erdős−Rényi random network with $p = 0.5$ (left two), $p = 0.1$ (middle two), and $p = 0.05$ (right two).

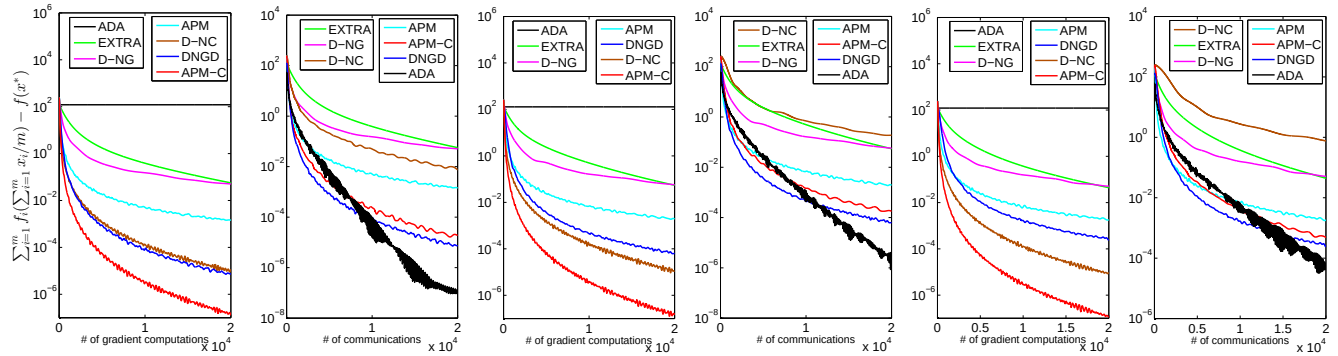


Fig. 2. Comparisons on the nonstrongly convex problem (32) and Erdős−Rényi random network with $p = 0.5$ (left two), $p = 0.1$ (middle two), and $p = 0.05$ (right two).

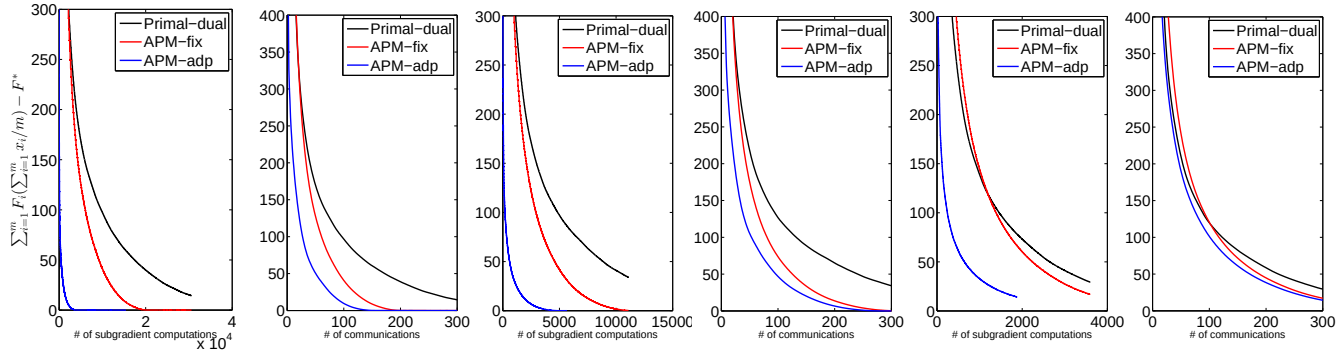


Fig. 3. Comparisons on the nonsmooth problem (33) and Erdős−Rényi random network with $p = 0.5$ (left two), $p = 0.1$ (middle two), and $p = 0.05$ (right two).

solution in the precision of ϵ due to the weakness of the regularization trick. From Figure 2, we can see that the value of $\frac{1}{\sqrt{1-\sigma_2(W)}}$ has less impact on the performance of APM-C than that in the strongly convex setting.

B. Non-smooth Problem

In this section, we follow [25] to test Algorithm 2 on the following decentralized linear Support Vector Machine (SVM) model

$$\min_{x\in\mathbb{R}^n} \sum_{i=1}^m f_i(x) \quad \text{with} \quad f_i(x) \equiv \max\{0, 1 - b_i A_i^T x\}. \quad (33)$$

The problem setting is similar to Section IV-A and the only difference is that we set $b_i = \text{Sign}(A_i^T x)$ for some $\mathbf{x}$ generated from the Gaussian distribution. We also consider the Erdős−Rényi random graph with $p = 0.05$, $p = 0.1$, and $p = 0.5$, respectively. We compare APM with the primal-dual method [11]. We test two different parameter settings for APM. For the first one, we follow Corollary 1 to set $\beta_0 = \frac{0.01}{\sqrt{1-\sigma_2(W)}}$, $T_k = \lceil k(1-\sigma_2(W))\rceil$, and $\eta_k = \frac{5000}{k^2\sqrt{1-\sigma_2(W)}}$, and name it APM with adaptive parameters (APM-adp). For the second one, we follow Theorem 3 to set $\beta_0 = \frac{0.01}{\sqrt{1-\sigma_2(W)}}$, $T_k =$

$\lceil K(1-\sigma_2(W))\rceil$, and $\eta_k = \frac{5000}{kK\sqrt{1-\sigma_2(W)}}$ with $K = 300$ and name it APM with fix parameters (APM-fix). For the primal-dual method, we set the number of inner iterations as $\lceil K(1-\sigma_2(W))\rceil$ and tune the best parameters of $\sigma = 1$ and $\eta = 0.5$ in [11, Alg 3]. Figure 3 plots the result. We can see that APM performs better than the primal-dual method, and APM-adp needs less communications and subgradient computations than APM-adp.

V. Conclusion

In this paper, we study the distributed accelerated gradient methods from the perspective of the accelerated penalty method with increasing penalty parameters. Two algorithms are proposed. The first algorithm achieves the optimal gradient computation complexities and near optimal communication complexities for both strongly convex and nonstrongly convex smooth distributed optimization. Our communication complexities are only worse by a factor of $\log\frac{1}{\epsilon}$ than the lower bounds. Our second algorithm obtains both the optimal subgradient computation and communication complexities for nonsmooth distributed optimization. Our APM-C is not suited to the network with large $\frac{1}{\sqrt{1-\sigma_2(W)}}$ for strongly convex problems, in which case the communication cost is high.

References

[1] J. Bazerque and G. Giannakis, "Distributed spectrum for cognitive radio networks by exploiting sparsity," *IEEE transactions on Signal Processing*, vol. 58, no. 3, pp. 1847–1862, 2010.

[2] S. Ram, V. Veeravalli, and A. Nedic, "Distributed non-autonomous power control through distributed convex optimization," in *International Conference on Computer Communications (INFOCOM)*, pp. 3001–3005, 2009.

[3] W. Ren, "Consensus based formation control strategies for multi-vehicle systems," in *American Control Conference (ACC)*, pp. 4237–4242, 2006.

[4] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao, "Optimal distributed online prediction using mini-batches," *Journal of Machine Learning Research*, vol. 13, pp. 165–202, 2012.

[5] P. Forero, A. Cano, and G. Giannakis, "Consensus-based distributed support vector machines," *Journal of Machine Learning Research*, vol. 59, pp. 1663–1707, 2010.

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

[7] G. Qu and N. Li, "Accelerated distributed Nesterov gradient descent," *IEEE Transactions on Automatic Control*, vol. 65, no. 6, pp. 2566–2581, 2020.

[8] D. Jakovetić, J. Xavier, and J. M. F. Moura, "Fast distributed gradient methods," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1131–1146, 2014.

[9] C. A. Uribe, S. Lee, A. Gasnikov, and A. Nedić, "A dual approach for optimal algorithms in distributed optimization over networks," *arXiv:1809.00710*, 2018.

[10] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic, Boston, 2004.

[11] 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*, vol. 20, pp. 1–31, 2019.

[12] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for smooth and strongly convex distributed optimization in networks," in *International Conference on Machine Learning (ICML)*, pp. 3027–3036, 2017.

[13] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for non-smooth distributed optimization in networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2740–2749, 2018.

[14] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[15] A. Nedić, "Asynchronous broadcast-based convex optimization over a network," *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1337–1351, 2011.

[16] S. Ram, A. Nedìc, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *Journal of Optimization Theory and Applications*, vol. 147, pp. 516–545, 2010.

[17] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes," in *IEEE Conference on Decision and Control (CDC)*, pp. 2055–2060, 2015.

[18] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1245–1260, 2018.

[19] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[20] W. Shi, Q. Ling, G. Wu, and W. Yin, "EXREA: An exact first-order algorithm for decentralized consensus optimization," *SIAM Journal Optimization*, vol. 25, no. 2, pp. 944–966, 2015.

[21] W. Shi, Q. Ling, G. Wu, and W. Yin, "A proximal gradient algorithm for decentralized composite optimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 23, pp. 6013–6023, 2015.

[22] A. Berahas, R. Bollapragada, N. Keskar, and E. Wei, "Balancing communication and computation in distributed optimization," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3141–3155, 2019.

[23] H. Terelius, U. Topcu, and R. Murray, "Decentralized multi-agent optimization via dual decomposition," *IFAC proceedings volumes*, vol. 44, no. 1, pp. 11245–11251, 2011.

[24] H. Yu and M. Neely, "On the convergence time of dual subgradient methdos for strongly convex programs," *IEEE Transactions on Automatic Control*, vol. 63, no. 4, pp. 1105–1112, 2018.

[25] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Mathematical Programming*, vol. 180, pp. 237–284, 2020.

[26] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast consensus by the alternating direction multipliers method," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5523–5537, 2011.

[27] W. Shi, Q. Ling, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 2, pp. 1750–1761, 2014.

[28] E. Wei and A. Ozdaglar, "On the $o(1/k)$ convergence of asynchronous distributed alternating direction method of multipliers," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 551–554, 2013.

[29] F. Iutzeler, P. Bianchi, P. Ciblat, and W. Hachem, "Explicit convergence rate of a distributed alternating direction method of multipliers," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 892–904, 2016.

[30] A. Makhdoumi and A. Ozdaglar, "Convergence rate of distributed ADMM over networks," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5082–5095, 2017.

[31] N. Aybat, Z. Wang, T. Lin, and S. Ma, "Distributed linearized alternating direction method of multipliers for composite convex consensus optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 1, pp. 5–20, 2018.

[32] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, pp. 120–145, 2011.

[33] M. Arioli and J. Scott, "Chebyshev acceleration of iterative refinement," *Numerical Algorithms*, vol. 66, no. 3, pp. 591–608, 2014.

[34] A. Nedić, A. Olshevsky, and M. Rabbat, "Network topology and communication-computation tradeoffs in decentralized optimization," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 953–976, 2018.

[35] M. Hong, D. Hajinezhad, and M.-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)*, pp. 1529–1538, 2017.

[36] D. Jakovetić, "A unification and generatliztion of exact distributed first order methods," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 1, pp. 31–46, 2019.

[37] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM Journal Optimization*, vol. 26, no. 3, pp. 1835–1854, 2016.

[38] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[39] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[40] J. Liu and A. S. Morse, "Accelerated linear iterations for distributed averaging," *Annual Reviews in Control*, vol. 35, no. 2, pp. 160–165, 2011.

[41] T. Zhang and H. Yu, "Average consensus for directed networks of multi-agent with time-varying delay," in *International Conference in Swarm Intelligence (ICSI)*, pp. 723–730, 2010.

[42] G. Lan, "Gradient sliding for composite optimization," *Mathematical Programming*, vol. 159, pp. 201–235, 2016.

[43] G. Lan and Y. Ouyang, "Accelerated gradient sliding for structured convex optimization," *preprint arXiv:1609.04905*, 2016.

[44] G. Lan and Y. Zhou, "Conditional gradient sliding for convex optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1379–1409, 2016.

[45] G. Lan and R. D. Monteiro, "Iteration-complexity of first-order penalty methods for convex programming," *Mathematical Programming*, vol. 138, pp. 115–139, 2013.

[46] I. Necoara, A. Patrascu, and F. Glineur, "Complexity of first-order inexact Lagrangian and penalty methods for conic convex programming," *Optimization Methods and Software*, vol. 34, no. 2, pp. 305–335, 2019.

[47] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$," *Doklady AN SSSR*, vol. 269, pp. 543–547, 1983.

[48] Y. Nesterov, "On an approach to the construction of optimal methods of minimization of smooth convex functions," *Èkonomika I Mateaticheskie Metody*, vol. 24, pp. 509–517, 1988.

[49] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, pp. 37–75, 2014.

[50] M. Schmidt, N. L. Roux, and F. R. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization," in *Advances in Neural Information Processing Systems (NIPS)*, pp. 1458–1466, 2011.

[51] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical Programming*, vol. 103, pp. 127–152, 2005.

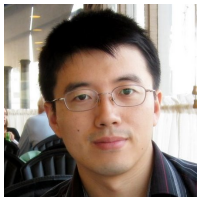
[52] H. Li and Z. Lin, "Accelerated alternating direction method of multipliers: an optimal $O(1/K)$ nonergodic analysis," *Journal of Scientific Computing*, vol. 79, pp. 671–699, 2019.

[53] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing markov chain on a graph," *SIAM Review*, vol. 46, no. 4, pp. 667–689, 2004.

[54] Y. Chow, W. Shi, T. Wu, and W. Yin, "Expander graph and communication-efficient decentralized optimization," in *Asilomar Conference on Signals, Systems and Computers (ACSSC)*, pp. 1715–1720, 2016.

Huan Li received his Ph.D. degree from Peking University in 2019. He is currently an Assistant Researcher at the Institute of Robotics and Automatic Information Systems, Nankai University. His current research interests include optimization and machine learning.

Cong Fang received his Ph.D. degree from Peking University in 2019. He is currently a Postdoctoral Researcher at Princeton University.His research interests include machine learning and optimization.



Wotao Yin received the B.S. degree in mathematics and applied mathematics from Nanjing University in 2001 and the M.S. and Ph.D. degrees in operations research from Columbia University in 2003 and 2006, respectively. From 2006 to 2013, he was an Assistant Professor and then an Associate Professor in the Department of Computational and Applied Mathematics, Rice University. Since 2013, he has been a Professor in the Department of Mathematics, University of California, Los Angeles, CA, USA. His current research interest is large-scale decentralized/distributed optimization.

Zhouchen Lin (M'00-SM'08-F'18) received the Ph.D. degree in Applied Mathematics from Peking University, in 2000. He is currently a Professor at Key Laboratory of Machine Perception (MOE), School of EECS, Peking University. His research interests include computer vision, image processing, machine learning, pattern recognition, and numerical optimization. He is an Associate Editor of IEEE Trans. Pattern Analysis and Machine Intelligence and International J. Computer Vision, an area chair of CVPR 2014/16/19/20/21, ICCV 2015, NIPS 2015/18/19/20, ICML 2020, AAAI 2019/20, IJCAI 2020/21 and ICLR 2021, and a Fellow of the IEEE and the IAPR.