2016

Asynchronous Distributed ADMM for Large-Scale Optimization—Part II: Linear Convergence Analysis and Numerical Performance

Tsung-Hui Chang
The Chinese University of Hong Kong

Wei-Cheng Lao
University of Minnesota - Twin Cities

Mingyi Hong
Iowa State University, mingyi@iastate.edu

Xiangfeng Wang
East China Normal University

Asynchronous Distributed ADMM for Large-Scale Optimization- Part II: Linear Convergence Analysis and Numerical Performance

Tsung-Hui Chang*, Wei-Cheng Liao§, Mingyi Hong† and Xiangfeng Wang‡

Abstract

The alternating direction method of multipliers (ADMM) has been recognized as a versatile approach for solving modern large-scale machine learning and signal processing problems efficiently. When the data size and/or the problem dimension is large, a distributed version of ADMM can be used, which is capable of distributing the computation load and the data set to a network of computing nodes. Unfortunately, a direct synchronous implementation of such algorithm does not scale well with the problem size, as the algorithm speed is limited by the slowest computing nodes. To address this issue, in a companion paper, we have proposed an asynchronous distributed ADMM (AD-ADMM) and studied its worst-case convergence conditions. In this paper, we further the study by characterizing the conditions under which the AD-ADMM achieves linear convergence. Our conditions as well as the resulting linear rates reveal the impact that various algorithm parameters, network delay and network size have on the algorithm performance. To demonstrate the superior time efficiency of the proposed AD-ADMM, we test the AD-ADMM on a high-performance computer cluster by solving a large-scale logistic regression problem.

Keywords– Distributed optimization, ADMM, Asynchronous, Consensus optimization

*Tsung-Hui Chang is the corresponding author. Address: School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, China 518172, E-mail: tsunghui.chang@ieee.org.

†Wei-Cheng Liao is with Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, USA, E-mail: mhong@umn.edu

†Mingyi Hong is with Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, 50011, USA, E-mail: mingyi@iastate.edu

‡Xiangfeng Wang is with Shanghai Key Lab for Trustworthy Computing, Software Engineering Institute, East China Normal University, Shanghai, 200062, China, E-mail: xfwang@sei.ecnu.edu.cn

I. INTRODUCTION

Consider the following optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \sum_{i=1}^{N} f_i(\boldsymbol{x}) + h(\boldsymbol{x}), \tag{1}$$

where each $f_i : \mathbb{R}^n \to \mathbb{R}$ is the cost function and $h : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is a non-smooth, convex regularization function. The regularization function is used for obtaining structured solutions (e.g., sparsity) and/or is an indicator function which enforces $\boldsymbol{x}$ to lie in a constraint set [2, Section 5]. Many important statistical learning problems can be formulated as problem (1), including, for example, the LASSO problem [3], logistic regression (LR) problem [4], support vector machine (SVM) [5] and the sparse principal component analysis (PCA) problem [6], to name a few.

Distributed optimization algorithms that can scale well with large-scale instances of (1) have drawn significant attention in recent years [2], [7]–[14]. Our interest in this paper lies in the distributed optimization method based on the alternating direction method of multipliers (ADMM) [2, Section 7.1.1]. The ADMM is a convenient approach of distributing the computation load of a very large-scale problem to a network of computing nodes. Specifically, consider a computer network with a star topology, where one master node coordinates the computation of a set of N distributed workers. Based on a consensus formulation, the distributed ADMM partitions the original problem into N subproblems, each of which contains either a small set of training samples or a subset of the learning parameters. At each iteration, the distributed workers solve the subproblems based on the local data and send the variable information to the master, who summarizes the variable information and broadcasts it back the workers. Through such iterative variable update and information exchange, the large-scale learning problem can be solved in a distributed and parallel manner.

The convergence conditions of the distributed ADMM have been extensively studied; see [2], [7], [15]–[20]. For example, for general convex problems, references [2], [7] showed that the ADMM is guaranteed to converge to an optimal solution and [15] showed that the ADMM has a worst-case $\mathcal{O}(1/k)$ convergence rate, where k is the iteration number. Considering non-convex problems with smooth f_i's, reference [16] presented conditions for which the distributed ADMM converges to the set of Karush-Kuhn-Tucker (KKT) points. For problems with strongly convex and smooth f_i's or problems satisfying certain error bound condition, references [17] and [21] respectively showed that the ADMM can even exhibit a linear convergence rate. References [18]–[20] also showed similar linear convergence conditions for some variants of distributed ADMM in a network with a general topology. However, the distributed ADMM in [2], [16] have assumed a synchronous network, where at each iteration, the master always waits until all

the workers report their variable information. Unfortunately, such synchronous protocol does not scale well with the problem size, as the algorithm speed is determined by the "slowest" workers. To improve the time efficiency, the works [22], [23] have generalized the distributed ADMM to an asynchronous network. Specifically, in the asynchronous distributed ADMM (AD-ADMM) proposed in [22], [23], the master does not necessarily wait for all the workers. Instead, the master updates its variable whenever it receives the variable information from a partial set of the workers. This prevents the master and speedy workers from spending most of the time waiting and consequently can improve the time efficiency of distributed optimization. Theoretically, it has been shown in [23] that the AD-ADMM is guaranteed to converge (to a KKT point) even for non-convex problem (1), under a bounded delay assumption only.

The contributions of this paper are twofold. Firstly, beyond the convergence analysis in [23], we further present the conditions for which the AD-ADMM can exhibit a linear convergence rate. Specifically, we show that for problem (1) with some structured convex f_i's (e.g., strongly convex), the augmented Lagrangian function of the AD-ADMM can decrease by a constant fraction in every iteration of the algorithm, as long as the algorithm parameters are chosen appropriately according to the network delay. We give explicit expressions on the linear convergence conditions and the linear rate, which illustrate how the algorithm and network parameters impact on the algorithm performance. To the best of our knowledge, our results are novel, and are by no means extensions of the existing analyses [17]–[21] for synchronous ADMM. Secondly, we present extensive numerical results to demonstrate the time efficiency of the AD-ADMM over its synchronous counterpart. In particular, we consider a large-scale LR problem and implement the AD-ADMM on a high-performance computer cluster. The presented numerical results show that the AD-ADMM significantly reduces the practical running time of distributed optimization.

Synopsis: Section II reviews the AD-ADMM in [23]. The linear convergence analysis is presented in Section III and the proofs are presented in Section IV. Numerical results are given in Section V and conclusions are drawn in Section VI.

II. Asynchronous Distributed ADMM

In this section, we review the AD-ADMM proposed in [23]. The distributed ADMM [2, Section 7.1.1] is derived based on the following consensus formulation of (1):

$$\min_{\substack{\boldsymbol{x}_0,\boldsymbol{x}_i\in\mathbb{R}^n,\\ i=1,\ldots,N}} \sum_{i=1}^{N} f_i(\boldsymbol{x}_i) + h(\boldsymbol{x}_0) \tag{2a}$$

$$\text{s.t. } \boldsymbol{x}_i = \boldsymbol{x}_0 \ \forall i \in \mathcal{V} \triangleq \{1,\ldots,N\}. \tag{2b}$$

By applying the standard ADMM [7] to problem (2), one obtains the following three simple steps: for iteration $k = 0, 1, \ldots,$ update

$$\boldsymbol{x}_0^{k+1} = \arg\min_{\boldsymbol{x}_0 \in \mathbb{R}^n} \left\{ h(\boldsymbol{x}_0) - \boldsymbol{x}_0^T \sum_{i=1}^N \boldsymbol{\lambda}_i^k + \frac{\rho}{2} \sum_{i=1}^N \|\boldsymbol{x}_i^k - \boldsymbol{x}_0\|^2 \right\}, \tag{3}$$

$$\boldsymbol{x}_i^{k+1} = \arg\min_{\boldsymbol{x}_i \in \mathbb{R}^n} \; f_i(\boldsymbol{x}_i) + \boldsymbol{x}_i^T \boldsymbol{\lambda}_i^k + \frac{\rho}{2} \|\boldsymbol{x}_i - \boldsymbol{x}_0^{k+1}\|^2 \quad \forall i \in \mathcal{V}, \tag{4}$$

$$\boldsymbol{\lambda}_i^{k+1} = \boldsymbol{\lambda}_i^k + \rho(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}) \quad \forall i \in \mathcal{V}. \tag{5}$$

As seen, the distributed ADMM is designed for a computing network with a star topology that consists of one master node and a set of N workers (see Fig. 1 in [23]). In particular, the master is responsible for optimizing the variable $\boldsymbol{x}_0$ by (3), while each worker i, $i \in \mathcal{V}$, takes charge of optimizing variables $\boldsymbol{x}_i$ and $\boldsymbol{\lambda}_i$ by (4) and (5), respectively. Once the master updates $\boldsymbol{x}_0$, it broadcasts $\boldsymbol{x}_0$ to the workers; each worker i then updates $(\boldsymbol{x}_i, \boldsymbol{\lambda}_i)$ based on the received $\boldsymbol{x}_0$, and sends the new $(\boldsymbol{x}_i, \boldsymbol{\lambda}_i)$ to the master. Through such iterative variable update and message exchange, problem (2) is solved in a fully parallel and distributed fashion.

However, to implement (3)-(5), the master and the workers have to be synchronized with each other. Specifically, according to (3), the master proceeds to update $\boldsymbol{x}_0$ only if it has received update-to-date $(\boldsymbol{x}_i, \boldsymbol{\lambda}_i)$ from all the workers. This implies that the optimization speed would be determined by the slowest worker in the network. This is in particular the case in a heterogeneous network where the workers experience different computation and communication delays, in which case the master and speedy workers would idle most of the time.

The distributed ADMM has been extended to an asynchronous network in [22], [23]. In the AD-ADMM, the master does not wait for all the workers, but updates the variable $\boldsymbol{x}_0$ as long as it receives variable information from a partial set of workers instead. This would greatly reduce the waiting time of the master, and improve the overall time efficiency of distributed optimization. The AD-ADMM is presented in Algorithm 1, which includes the algorithmic steps of the master and those of the workers. Here, we denote k as the iteration number of the master (i.e., the number of times for which the master updates $\boldsymbol{x}_0$), and assume that, at each iteration k, the master receives variable information from workers in the set $\mathcal{A}_k \subseteq \mathcal{V} \triangleq \{1, \ldots, N\}$. Worker i is said to be "arrived" at iteration k if $i \in \mathcal{A}_k$ and unarrived otherwise. Notation $\mathcal{A}_k^c$ denotes the complementary set of $\mathcal{A}_k$, i.e., $\mathcal{A}_k \cap \mathcal{A}_k^c = \emptyset$ and $\mathcal{A}_k \cup \mathcal{A}_k^c = \mathcal{V}$. Moreover, variables d_i's are used to count the numbers of delayed iterations of the workers. The variables ρ and γ are two penalty parameters.

In the AD-ADMM, the master inevitably uses delayed and old variable information for updating $\boldsymbol{x}_0$. As shown in step 4 of Algorithm of the Master, to ensure the used variable information not too stale, the master would wait until it receives the update-to-date $(\boldsymbol{x}_i, \boldsymbol{\lambda}_i)$ from all the workers that have $d_i \geq \tau - 1$, if any (so all the workers $i \in \mathcal{A}_k^c$ must have $d_i < \tau - 1$). This condition guarantees that the variable information is at most τ iterations old, and is known as the partially asynchronous model [7]:

Assumption 1 (Bounded delay) *Let $\tau \geq 1$ be a maximum tolerable delay. For all $i \in \mathcal{V}$ and iteration k, it must be that $i \in \mathcal{A}_k \cup \mathcal{A}_{k-1} \cdots \cup \mathcal{A}_{k-\tau+1}$.*

In [23, Theorem 1], we have shown that under Assumption 1, some smoothness conditions on the cost functions f_i's (see [23, Assumption 2]) and for sufficiently large ρ and γ, the AD-ADMM in Algorithm 1 is provably convergent to the set of KKT points of problem (2). Notably, this convergence property holds even for non-convex f_i's. In the next section, we focus on convex f_i's, and further characterize the linear convergence conditions of the AD-ADMM.

III. Linear Convergence Rate Analysis

In this section, we show that the AD-ADMM can achieve linear convergence for some structured convex functions. We first make the following convex assumption on problem (1) (or equivalently, problem (2)).

Assumption 2 *Each function f_i is a proper closed convex function and is continuously differentiable; each gradient ∇f_i is Lipschitz continuous with a Lipschitz constant $L > 0$; the function h is convex (not necessarily smooth). Moreover, problem* (1) *is bounded below, i.e., $F^\star > -\infty$ where $F^\star$ denotes the optimal objective value of problem* (1)*.*

Assumption 2 is the same as [23, Assumption 2], except that f_i's are assumed convex here. Given this convex property, it is well known that the augmented Lagrangian function, i.e.,

$$\mathcal{L}_\rho(\boldsymbol{x}^k, \boldsymbol{x}_0^k, \boldsymbol{\lambda}^k) = \sum_{i=1}^N f_i(\boldsymbol{x}_i^k) + h(\boldsymbol{x}_0^k) + \sum_{i=1}^N (\boldsymbol{\lambda}_i^k)^T(\boldsymbol{x}_i^k - \boldsymbol{x}_0^k) + \frac{\rho}{2}\sum_{i=1}^N \|\boldsymbol{x}_i^k - \boldsymbol{x}_0^k\|^2, \tag{12}$$

would converge to $F^\star$ whenever the iterates $(\{\boldsymbol{x}_i^k\}_{i=1}^N, \boldsymbol{x}_0^k, \{\boldsymbol{\lambda}_i^k\}_{i=1}^N)$ approaches the optimal solution of problem (2). Therefore, our analysis is based on characterizing how $\mathcal{L}_\rho(\boldsymbol{x}^k, \boldsymbol{x}_0^k, \boldsymbol{\lambda}^k)$ can converge to $F^\star$ linearly. Let us define

$$\triangle_k \triangleq \mathcal{L}_\rho(\boldsymbol{x}^k, \boldsymbol{x}_0^k, \boldsymbol{\lambda}^k) - F^\star. \tag{13}$$

It has been shown in [23, Lemma 3] that $\triangle_k \geq 0$ for all k as long as $\rho \geq L$.

In the ensuing analysis, we consider two types of structured convex cost functions, respectively described in the following two assumptions.

Assumption 3 *For all $i \in \mathcal{V}$, each function f_i is strongly convex with modulus $\sigma^2 > 0$.*

Assumption 4 *Each function $f_i(\boldsymbol{x}) = g_i(\boldsymbol{A}_i\boldsymbol{x})$, $\forall i \in \mathcal{V}$, where $g_i : \mathbb{R}^m \to \mathbb{R}$ is a strongly convex function with modulus $\sigma^2 > 0$ and $\boldsymbol{A}_i \in \mathbb{R}^{m\times n}$ is a nonzero matrix with arbitrary rank. Moreover, $h(\boldsymbol{x}) = 0$.*

Note that in Assumption 4 matrix $\boldsymbol{A}_i$ can have an arbitrary rank, so $f_i(\boldsymbol{x})$ is not necessarily strongly convex with respect to $\boldsymbol{x}$. Interestingly, such structured cost function appears in many machine learning problems, for example, the least squared problem and the logistic regression problem [5].

Let us first consider the strongly convex case. Under Assumption 3, the linear convergence conditions of the AD-ADMM are given by the following theorem.

Theorem 1 *Suppose that Assumptions 1, 2 and 3 hold true. Moreover, assume that there exists a constant $S \in [1, N]$ such that $|\mathcal{A}_k| < S$ for all k and that*

$$\rho \geq \max\left\{\frac{(1+L^2)+\sqrt{(1+L^2)^2+8L^2\alpha(\tau)}}{2}, \sigma^2+\frac{1}{8N}\right\}, \tag{14}$$

$$\gamma \geq \max\left\{\beta(\rho,\tau)-\frac{N\rho}{2}+1, 8N(\rho-\sigma^2)\right\}, \tag{15}$$

where $\alpha(\tau) \triangleq 1+\frac{2+2^\tau(\tau-1)}{1+8N\sigma^2}$ and $\beta(\rho,\tau) \triangleq 2(\tau-1)[(\frac{(1+\rho^2)S+S/N}{2})(2^{\tau-1}-1)+(4^{\tau-1}-1)]$. Then, the iterates generated by (6), (7) *and* (9) *satisfy*

$$0 \leq \triangle_{k+1} \leq \left(\frac{1}{1+\frac{1}{\delta\gamma}}\right)^{k+1}\triangle_0, \tag{16}$$

where δ is a constant satisfying

$$\delta \geq \max\left\{1, \frac{\rho N+\gamma}{\sigma^2 N}-1\right\}. \tag{17}$$

Theorem 1 asserts that, for problem (2) with strongly convex f_i's, the augmented Lagrange function can decrease linearly to zero, as long as ρ and γ are large enough (exponentially increasing with τ). Equation (16) also implies that the linear rate would decrease with the delay τ and the number of workers in the worst case.

Analogous to Theorem 1, the following theorem shows that the AD-ADMM can achieve linear convergence under Assumption 4.

Theorem 2 *Suppose that Assumptions 1, 2 and 4 hold true. Moreover, assume that there exists a constant* $S \in [1, N]$ *such that* $|\mathcal{A}_k| < S$ *for all* k *and that*

$$\rho \geq \max\left\{\frac{(1+L^2)+\sqrt{(1+L^2)^2+8L^2\alpha(\tau)}}{2}, \sigma^2+\frac{1}{8N}\right\},$$
$$\gamma \geq \max\left\{\beta(\rho,\tau)-\frac{N\rho}{2}+1, 8N(\rho-\sigma^2/c)+4N\sigma^2\right\},$$

for some constant $c > 0$*. Then, the iterates generated by* (6)*,* (7) *and* (9) *satisfy* (16) *with* δ *satisfying*

$$\delta \geq \max\left\{1, \frac{\rho N+\gamma}{N\sigma^2/c}-1\right\}.$$

Since it has been known that the (synchronous) distributed ADMM [17]–[21] can converge linearly given the same structured cost functions in Assumption 3 and Assumption 4, the convergence results presented above demonstrate that the linear convergence property can be preserved in the asynchronous network. We remark that (14) and (15) are sufficient conditions only. In practice, the AD-ADMM could still exhibit a linear convergence rate without exactly satisfying these conditions.

The proofs of Theorem 1 and Theorem 2 are presented in the next section. The readers who are more interested in the numerical performance of the AD-ADMM may jump to Section V.

IV. Proofs of Theorems

A. Preliminaries and Key Lemmas

Let us present some basic inequalities that will be used frequently in the ensuing analysis and key lemmas for proving Theorem 1 and Theorem 2.

We will frequently use the following inequality due to Jensen's inequality: for any $\boldsymbol{a}_i$, $i = 1, \ldots, M$,

$$\|\textstyle\sum_{i=1}^M \boldsymbol{a}_i\|^2 \leq M \sum_{i=1}^M \|\boldsymbol{a}_i\|^2. \tag{18}$$

Moreover, for any $\boldsymbol{a}$, $\boldsymbol{b}$ and $\delta > 0$,

$$\|\boldsymbol{a}+\boldsymbol{b}\|^2 \leq (1+\delta)\|\boldsymbol{a}\|^2 + (1+\frac{1}{\delta})\|\boldsymbol{b}\|^2. \tag{19}$$

The equality is also known to be true: for any vectors $\boldsymbol{a}$, $\boldsymbol{b}$, $\mathbf{c}$ and $\boldsymbol{d}$,

$$\begin{aligned}(\boldsymbol{a}-\boldsymbol{b})^T(\mathbf{c}-\boldsymbol{d}) = &\frac{1}{2}\|\boldsymbol{a}-\boldsymbol{d}\|^2 - \frac{1}{2}\|\boldsymbol{a}-\mathbf{c}\|^2 \\ &+ \frac{1}{2}\|\boldsymbol{b}-\mathbf{c}\|^2 - \frac{1}{2}\|\boldsymbol{b}-\boldsymbol{d}\|^2.\end{aligned} \tag{20}$$

We follow [23, Algorithm 3] to write Algorithm 1 from the master's point of view as follows:

$$\boldsymbol{x}_i^{k+1}=\begin{cases}\arg\min\limits_{\boldsymbol{x}_i\in\mathbb{R}^n}\left\{f_i(\boldsymbol{x}_i)+\boldsymbol{x}_i^T\boldsymbol{\lambda}_i^k+\frac{\rho}{2}\|\boldsymbol{x}_i-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2\right\}, & \forall i\in\mathcal{A}_k\\ \boldsymbol{x}_i^k & \forall i\in\mathcal{A}_k^c\end{cases}, \tag{21}$$

$$\boldsymbol{\lambda}_i^{k+1}=\begin{cases}\boldsymbol{\lambda}_i^k+\rho(\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\bar{k}_i+1}) & \forall i\in\mathcal{A}_k\\ \boldsymbol{\lambda}_i^k & \forall i\in\mathcal{A}_k^c\end{cases}, \tag{22}$$

$$\begin{aligned}\boldsymbol{x}_0^{k+1}=&\arg\min_{\boldsymbol{x}_0\in\mathbb{R}^n}\Big\{h(\boldsymbol{x}_0)-\boldsymbol{x}_0^T\textstyle\sum_{i=1}^N\boldsymbol{\lambda}_i^{k+1}\\ &+\tfrac{\rho}{2}\textstyle\sum_{i=1}^N\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0\|^2+\tfrac{\gamma}{2}\|\boldsymbol{x}_0-\boldsymbol{x}_0^k\|^2\Big\}.\end{aligned} \tag{23}$$

Here, index $\bar{k}_i$ in (21) and (22) represents the last iteration number before iteration k for which worker $i\in\mathcal{A}_k$ is arrived, i.e., $i\in\mathcal{A}_{\bar{k}_i}$. Under Assumption 1, it must hold

$$k-\tau\leq\bar{k}_i<k\quad\forall k. \tag{24}$$

Furthermore, for workers $i\in\mathcal{A}_k^c$, let us denote $\widetilde{k}_i$ as the last iteration number before iteration k for which worker i is arrived, i.e., $i\in\mathcal{A}_{\widetilde{k}_i}$. Then, under Assumption 1, it must hold

$$k-\tau<\widetilde{k}_i<k\quad\forall k. \tag{25}$$

In addition, denote $\widehat{k}_i$ ($\widetilde{k}_i-\tau\leq\widehat{k}_i<\widetilde{k}_i$) as the last iteration number before iteration $\widetilde{k}_i$ for which worker $i\in\mathcal{A}_{\widetilde{k}_i}$ is arrived, i.e., $i\in\mathcal{A}_{\widehat{k}_i}$. Then by (21) and (22), for all workers $i\in\mathcal{A}_k^c$, we must have

$$\boldsymbol{x}_i^{\widetilde{k}_i+1}=\boldsymbol{x}_i^{\widetilde{k}_i+2}=\cdots=\boldsymbol{x}_i^k=\boldsymbol{x}_i^{k+1}, \tag{26}$$

$$\boldsymbol{\lambda}_i^{\widetilde{k}_i+1}=\boldsymbol{\lambda}_i^{\widetilde{k}_i+2}=\cdots=\boldsymbol{\lambda}_i^k=\boldsymbol{\lambda}_i^{k+1}, \tag{27}$$

Since $i\in\mathcal{A}_{\widetilde{k}_i}$ for all $i\in\mathcal{A}_k^c$ and by (26)-(27), we can equivalently write (21) and (22) for all $i\in\mathcal{A}_k^c$ as

$$\begin{aligned}\boldsymbol{x}_i^{k+1}&=\boldsymbol{x}_i^{\widetilde{k}_i+1}\\ &=\arg\min_{\boldsymbol{x}_i}\ f_i(\boldsymbol{x}_i)+\boldsymbol{x}_i^T\boldsymbol{\lambda}_i^{\widetilde{k}_i}+\tfrac{\rho}{2}\|\boldsymbol{x}_i-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2,\end{aligned} \tag{28}$$

$$\begin{aligned}\boldsymbol{\lambda}_i^{k+1}=\boldsymbol{\lambda}_i^{\widetilde{k}_i+1}&=\boldsymbol{\lambda}_i^{\widetilde{k}_i}+\rho(\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_0^{\widehat{k}_i+1})\\ &=\boldsymbol{\lambda}_i^{\widetilde{k}_i}+\rho(\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\widehat{k}_i+1}).\end{aligned} \tag{29}$$

Based on these notations, we have shown in [23, Eqn. (33)] that the following lemma is true.

Lemma 1 *Suppose that Assumption 2 holds and $\rho \geq L$. Then, for all $k = 0, 1, \ldots,$*

$$\begin{aligned} 0 \leq \triangle_{k+1} \leq \triangle_k &+ \left(\frac{1+\rho/\epsilon}{2}\right)\sum_{i\in\mathcal{A}_k} \|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\bar{k}_i+1}\|^2 \\ &- \left(\frac{2\gamma + N\rho}{2}\right)\|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2 + \left(\frac{L^2 + (\epsilon-1)\rho}{2} + \frac{L^2}{\rho}\right)\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i^k\|^2, \end{aligned} \tag{30}$$

where $\epsilon \in (0, 1)$ is a constant.

In particular, (30) is the same as [23, Eqn. (33)] except that here we have assumed convex f_i's. Lemma 1 shows how the gap between the augmented Lagrangian function $\mathcal{L}_\rho(\boldsymbol{x}^{k+1}, \boldsymbol{x}_0^{k+1}, \boldsymbol{\lambda}^{k+1})$ and the optimal objective value $F^\star$ evolves with the iteration number k. Notice that it follows from [23, Lemma 3] that $\triangle_{k+1} \geq 0$ for all k if $\rho \geq L$. As will be seen shortly, Lemma 1 is crucial in the linear convergence analysis.

Similar to [23, Lemma 3], we next need to bound the error terms, e.g., $(\frac{1+\rho^2}{2})\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\bar{k}_i+1}\|^2$ in (30), which is caused by asynchrony of the network. Here, we present a more general result for the latter analysis.

Lemma 2 *Let $\eta > 0$ and $j - \nu \leq j_i < j$ where $\nu \in \mathbb{Z}_{++}$, $j_i \in \mathbb{Z}_+$ and $j \in \{0, 1, \ldots, k\}$. Moreover, let $\mathcal{N}_j \subset \mathcal{V}$ be any index subset satisfying $|\mathcal{N}_j| \leq \bar{N}$ for some constant $\bar{N} \in (1, N]$. Then, the following inequality holds true*

$$\sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{N}_j}\|\boldsymbol{x}_0^j - \boldsymbol{x}_0^{j_i+1}\|^2 \leq (\nu-1)\bar{N}\sum_{j=0}^{k-1}\eta^{j+1}\left(\frac{\eta^{\nu-1}-1}{\eta-1}\right)\|\boldsymbol{x}_0^j - \boldsymbol{x}_0^{j+1}\|^2. \tag{31}$$

Proof: See Appendix B. ■

Now let us consider Assumption 3. For strongly convex $f_i's$, it is known that the following first-order condition holds [24]: $\forall \boldsymbol{x}, \boldsymbol{y}$,

$$f_i(\boldsymbol{y}) \geq f_i(\boldsymbol{x}) + (\nabla f_i(\boldsymbol{x}))^T(\boldsymbol{y} - \boldsymbol{x}) + \frac{\sigma^2}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2. \tag{32}$$

Based on this property, we can bound $\triangle_{k+1}$ as follows.

Lemma 3 *Suppose that Assumptions 2 and 3 hold and $\rho \geq \sigma^2$. If $\gamma \geq 8N(\rho - \sigma^2)$ and δ satisfies* (17)*, then it holds that*

$$\frac{1}{\gamma\delta}\triangle_{k+1} \leq \frac{L^2}{4\rho^2 N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i^k\|^2$$

$$+\frac{L^2}{4\rho^2 N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2+\frac{1}{2N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2$$
$$+\frac{1}{2N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2+\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2. \tag{33}$$

Instead, if $\gamma=0$ *and* $\delta\geq\max\{\rho/\sigma^2-1,1\}$*, then it holds*

$$\left(\frac{1}{4(\rho-\sigma^2)N\delta}\right)\triangle_{k+1}\leq\frac{L^2}{2\rho^2 N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2$$
$$+\frac{L^2}{2\rho^2 N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2+\frac{1}{N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2$$
$$+\frac{1}{N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2+\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2. \tag{34}$$

Proof: See Appendix C. ■

B. Proof of Theorem 1

We use the lemmas above to prove Theorem 1. Denote $\eta\triangleq 1+\frac{1}{\delta\gamma}$. By summing (30) and (33), we obtain

$$\triangle_{k+1}\leq\frac{1}{\eta}\triangle_k+\frac{1}{\eta}\bigg[\left(\frac{L^2+(\epsilon-1)\rho+\frac{L^2}{2\rho^2N}}{2}+\frac{L^2}{\rho}\right)\sum_{i=1}^N\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2$$
$$-\left(\frac{2\gamma+N\rho}{2}-1\right)\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2+\frac{1}{2N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2$$
$$+\frac{L^2}{4\rho^2N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|_2^2+\left(\frac{1+\rho/\epsilon}{2}+\frac{1}{2N}\right)\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2\bigg]. \tag{35}$$

Here, we have used the fact of $\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2=\sum_{i=1}^N\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2$ as $\boldsymbol{x}_i^{k+1}=\boldsymbol{x}_i^k\ \forall i\in\mathcal{A}_k^c$. By taking the telescoping sum of (35), we further obtain

$$\triangle_{k+1}\leq\frac{1}{\eta^{k+1}}\triangle_0$$
$$+\frac{1}{\eta}\bigg[\left(\frac{L^2+(\epsilon-1)\rho+\frac{L^2}{2\rho^2N}+\frac{2L^2}{\rho}}{2}\right)\sum_{\ell=0}^k\frac{1}{\eta^\ell}\sum_{i=1}^N\|\boldsymbol{x}_i^{k-\ell+1}-\boldsymbol{x}_i^{k-\ell}\|^2-\left(\frac{2\gamma+N\rho}{2}-1\right)\sum_{\ell=0}^k\frac{1}{\eta^\ell}\|\boldsymbol{x}_0^{k-\ell+1}-\boldsymbol{x}_0^{k-\ell}\|^2$$
$$+\left(\frac{1+\rho/\epsilon}{2}+\frac{1}{2N}\right)\underbrace{\sum_{\ell=0}^k\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{\overline{(k-\ell)}_i+1}\|^2}_{(36a)}+\frac{1}{2N}\underbrace{\sum_{\ell=0}^k\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}^c}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{\widehat{(k-\ell)}_i+1}\|^2}_{(36b)}$$
$$+\frac{L^2}{4\rho^2N}\underbrace{\sum_{\ell=0}^k\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}^c}\|\boldsymbol{x}_i^{\widetilde{(k-\ell)}_i+1}-\boldsymbol{x}_i^{\widetilde{(k-\ell)}_i}\|_2^2}_{(36c)}\bigg]. \tag{36}$$

The three terms (36a), (36b), and (36c) in the right hand side (RHS) of (36) can respectively be bounded as follows, using Lemma 2. Consider the change of variable $k-\ell=j$. Then, we have the following chain for (36a):

$$\begin{aligned}(36a) &= \sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{\overline{(k-\ell)}_i+1}\|^2\\ &= \frac{1}{\eta^k}\sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{A}_j}\|\boldsymbol{x}_0^{j}-\boldsymbol{x}_0^{\bar{j}_i+1}\|^2\\ &\le \frac{1}{\eta^k}S(\tau-1)\sum_{j=0}^{k-1}\eta^{j+1}\left(\frac{\eta^{\tau-1}-1}{\eta-1}\right)\|\boldsymbol{x}_0^{j}-\boldsymbol{x}_0^{j+1}\|^2\\ &= S(\tau-1)\eta\left(\frac{\eta^{\tau-1}-1}{\eta-1}\right)\sum_{\ell=1}^{k}\frac{1}{\eta^\ell}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{k-\ell+1}\|^2,\end{aligned}\tag{37}$$

where the inequality is obtained by applying (31) with $\nu=\tau$, $\mathcal{N}_j=\mathcal{A}_j$, $\bar{N}=S$, and $j_i=\bar{j}_i$ which satisfies $j-\tau\le\bar{j}_i<j$ (see (24)); to obtain the last equality, the change of variable $k-\ell=j$ is applied again.

Analogously, by applying (31) with $\nu=2\tau-1$, $\mathcal{N}_j=\mathcal{A}_j^c$, $\bar{N}=N$, and $j_i=\widehat{j}_i$ (which satisfies $j-2\tau+1\le\widehat{j}_i<j$ since $\widetilde{j}_i-\tau\le\widehat{j}_i<\widetilde{j}_i$ and $j-\tau<\widetilde{j}_i<j$ by (24) and (25)), one can bound (36b) as

$$\begin{aligned}(36b) &= \sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}^c}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{\widehat{(k-\ell)}_i+1}\|^2\\ &= \frac{1}{\eta^k}\sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{A}_j^c}\|\boldsymbol{x}_0^{j}-\boldsymbol{x}_0^{\widehat{j}_i+1}\|^2\\ &\le \frac{1}{\eta^k}2N(\tau-1)\sum_{j=0}^{k-1}\eta^{j+1}\left(\frac{\eta^{2(\tau-1)}-1}{\eta-1}\right)\|\boldsymbol{x}_0^{j}-\boldsymbol{x}_0^{j+1}\|^2\\ &= 2N(\tau-1)\eta\left(\frac{\eta^{2(\tau-1)}-1}{\eta-1}\right)\sum_{\ell=1}^{k}\frac{1}{\eta^\ell}\|\boldsymbol{x}_0^{k-\ell}-\boldsymbol{x}_0^{k-\ell+1}\|^2.\end{aligned}\tag{38}$$

The term (36c) can be bounded as follows

$$\begin{aligned}(36c) &= \sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\sum_{i\in\mathcal{A}_{k-\ell}^c}\|\boldsymbol{x}_i^{\widetilde{(k-\ell)}_i+1}-\boldsymbol{x}_i^{\widetilde{(k-\ell)}_i}\|^2\\ &= \frac{1}{\eta^k}\sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{A}_j^c}\|\boldsymbol{x}_i^{\widetilde{j}_i+1}-\boldsymbol{x}_i^{\widetilde{j}_i}\|^2\\ &= \frac{1}{\eta^k}\sum_{j=0}^{k}\sum_{i\in\mathcal{A}_j^c}\eta^{j-\widetilde{j}_i-1}\eta^{\widetilde{j}_i+1}\|\boldsymbol{x}_i^{\widetilde{j}_i+1}-\boldsymbol{x}_i^{\widetilde{j}_i}\|^2\end{aligned}$$

$$\leq \eta^{\tau-2}\frac{1}{\eta^k}\sum_{j=0}^{k}\sum_{i\in\mathcal{A}_j^c}\eta^{\widetilde{j}_i+1}\|\boldsymbol{x}_i^{\widetilde{j}_i+1}-\boldsymbol{x}_i^{\widetilde{j}_i}\|^2$$

$$\leq \eta^{\tau-2}(\tau-1)\frac{1}{\eta^k}\sum_{i=1}^{N}\sum_{j=0}^{k}\eta^{j+1}\|\boldsymbol{x}_i^{j+1}-\boldsymbol{x}_i^{j}\|^2$$

$$= \eta^{\tau-1}(\tau-1)\sum_{i=1}^{N}\sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\|\boldsymbol{x}_i^{k-\ell+1}-\boldsymbol{x}_i^{k-\ell}\|^2, \tag{39}$$

where, in the first inequality, we have used the fact of $j-\tau+1\leq\widetilde{j}_i<j$ from (25). To show the second inequality, notice that for any $i\in\mathcal{A}_j^c$, it also satisfies $i\in\mathcal{A}_\ell^c$ for $\ell=\widetilde{j}_i+1,\ldots,j$. So, $\widetilde{j}_i=\widetilde{\ell}_i$ for $\ell=\widetilde{j}_i+1,\ldots,j$. Since $j-\tau<\widetilde{j}_i<j$, each $\eta^{\widetilde{j}+1}\|\boldsymbol{x}_i^{\widetilde{j}+1}-\boldsymbol{x}_i^{\widetilde{j}_i}\|^2$ appears no more than $\tau-1$ times in the summation $\sum_{j=0}^{k}\sum_{i\in\mathcal{A}_j^c}\eta^{\widetilde{j}+1}\|\boldsymbol{x}_i^{\widetilde{j}+1}-\boldsymbol{x}_i^{\widetilde{j}_i}\|^2$.

By substituting (39), (38) and (37) into (36), we obtain

$$\begin{aligned}\triangle_{k+1}\leq{}& \frac{1}{\eta^{k+1}}\triangle_0\\
&+\frac{1}{\eta}\Bigg[\bigg(\frac{1+\rho/\epsilon}{2}+\frac{1}{2N}\bigg)S(\tau-1)\eta\bigg(\frac{\eta^{\tau-1}-1}{\eta-1}\bigg)\\
&\quad+(\tau-1)\eta\bigg(\frac{\eta^{2(\tau-1)}-1}{\eta-1}\bigg)\\
&\quad+1-\bigg(\frac{2\gamma+N\rho}{2}\bigg)\Bigg]\sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\|\boldsymbol{x}_0^{k-\ell+1}-\boldsymbol{x}_0^{k-\ell}\|^2\\
&+\frac{1}{\eta}\Bigg[\bigg(\frac{L^2+(\epsilon-1)\rho+\frac{L^2}{2\rho^2N}+\frac{2L^2}{\rho}}{2}\bigg)\\
&+\eta^{\tau-1}(\tau-1)\frac{L^2}{4\rho^2N}\Bigg]\sum_{i=1}^{N}\sum_{\ell=0}^{k}\frac{1}{\eta^\ell}\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k-\ell+1}-\boldsymbol{x}_i^{k-\ell}\|^2.\end{aligned} \tag{40}$$

Let $\epsilon=1/\rho$. Therefore, we see that (16) is true if

$$\begin{aligned}\gamma\geq{}&(\tau-1)\eta\Bigg[\bigg(\frac{S(1+\rho^2)+S/N}{2}\bigg)\bigg(\frac{\eta^{\tau-1}-1}{\eta-1}\bigg)\\
&+\bigg(\frac{\eta^{2(\tau-1)}-1}{\eta-1}\bigg)\Bigg]-\frac{N\rho}{2}+1,\end{aligned} \tag{41}$$

$$\rho\geq(1+L^2)+\frac{2L^2}{\rho}+\frac{L^2}{2\rho^2N}\bigg(1+\eta^{\tau-1}(\tau-1)\bigg). \tag{42}$$

Let $\rho\geq\frac{1}{8N}+\sigma^2$. Then (42) holds true if

$$\rho\geq(1+L^2)+\frac{2L^2}{\rho}\bigg(1+\frac{2+2\eta^{\tau-1}(\tau-1)}{1+8N\sigma^2}\bigg). \tag{43}$$

Moreover, since $\gamma \geq 8N(\rho - \sigma^2)$ and $\delta > 1$, we see that η has an upper bound

$$\eta = 1 + \frac{1}{\delta\gamma} < 1 + \frac{1}{8N(\rho - \sigma^2)} < 2. \tag{44}$$

Therefore, (14) and (15) are sufficient conditions for (43) and (41), respectively. The proof is thus complete. ■

C. Proof of Theorem 2

The key is to build a similar result as Lemma 3 under Assumption 4. Now, consider Assumption 4. Let $\boldsymbol{x}^\star$ be an optimal solution to (1), and let

$$\boldsymbol{y}_i^\star = \boldsymbol{A}_i \boldsymbol{x}^\star,\ i = 1, \ldots, N.$$

Then, $(\boldsymbol{y}_1^\star, \ldots, \boldsymbol{y}_N^\star)$ is unique since g_i's are strongly convex. So, the optimal solution set to (2) can be defined as

$$\mathcal{X}^\star = \Bigg\{ (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) \mid \begin{bmatrix} \boldsymbol{y}_1^\star \\ \vdots \\ \boldsymbol{y}_N^\star \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1 \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{A}_N \boldsymbol{x}_N \end{bmatrix}, \\ \boldsymbol{x}_i = \boldsymbol{x}_0,\ i = 1, \ldots, N \Bigg\}. \tag{45}$$

Let $\mathbf{1}_{N+1} \otimes \mathcal{P}^\star(\hat{\boldsymbol{x}})$ be the projection point of $\hat{\boldsymbol{x}} \triangleq (\boldsymbol{x}_0^T, \boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T)^T$ onto $\mathcal{X}^\star$, where $\otimes$ denotes the Kronecker product. It can be shown that the following lemma is true.

Lemma 4 *Under Assumption 4, for any $\hat{\boldsymbol{x}} \in \mathbb{R}^{n(N+1)}$, it holds that*

$$\begin{aligned} \sum_{i=1}^N f_i(\mathcal{P}^\star(\hat{\boldsymbol{x}})) \geq \sum_{i=1}^N f_i(\boldsymbol{x}_i) + \sum_{i=1}^N (\nabla f_i(\boldsymbol{x}_i))^T (\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{x}_i) \\ + \sum_{i=1}^N \frac{\sigma^2}{2c} \|\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{x}_i\|^2 + \frac{\sigma^2}{2c} \|\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{x}_0\|^2 \\ - \frac{\sigma^2}{2} \sum_{i=1}^N \|\boldsymbol{x}_i - \boldsymbol{x}_0\|^2, \end{aligned} \tag{46}$$

for some finite constant $c > 0$.

Proof: See Appendix D. ■

Lemma 4 implies that the structured f_i's in Assumption 4 own an analogous property as the strongly convex functions in (32). Based on Lemma 4, the next lemma shows that one can still bound $\triangle_{k+1}$ as in Lemma 3 under Assumption 4.

Lemma 5 *Suppose that Assumptions 2 and 4 hold, and assume that* $\gamma \geq 8N(\rho - \sigma^2/c) + 4N\sigma^2$ *and* δ *satisfies*

$$\delta \geq \max\left\{1, \frac{\rho N + \gamma}{N\sigma^2/c} - 1\right\}. \tag{47}$$

Then, (33) *holds true. Instead, if* $\gamma = 0$ *and* $\delta \geq \max\{(c\rho)/\sigma^2 - 1, 1\}$*, then*

$$\begin{aligned}\frac{\triangle_{k+1}}{2N[2(\rho - \sigma^2/c)\delta + \sigma^2]} \leq \frac{L^2}{2\rho^2 N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i^k\|^2 \\ + \frac{L^2}{2\rho^2 N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1} - \boldsymbol{x}_i^{\widetilde{k}_i}\|^2 + \frac{1}{N}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\bar{k}_i+1}\|^2 \\ + \frac{1}{N}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\widehat{k}_i+1}\|^2 + \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2.\end{aligned} \tag{48}$$

Proof: See Appendix E. ■

Given Lemma 5, Theorem 2 can be proved by following exactly the same steps as for Theorem 1 in Section IV-B. The details are omitted here. ■

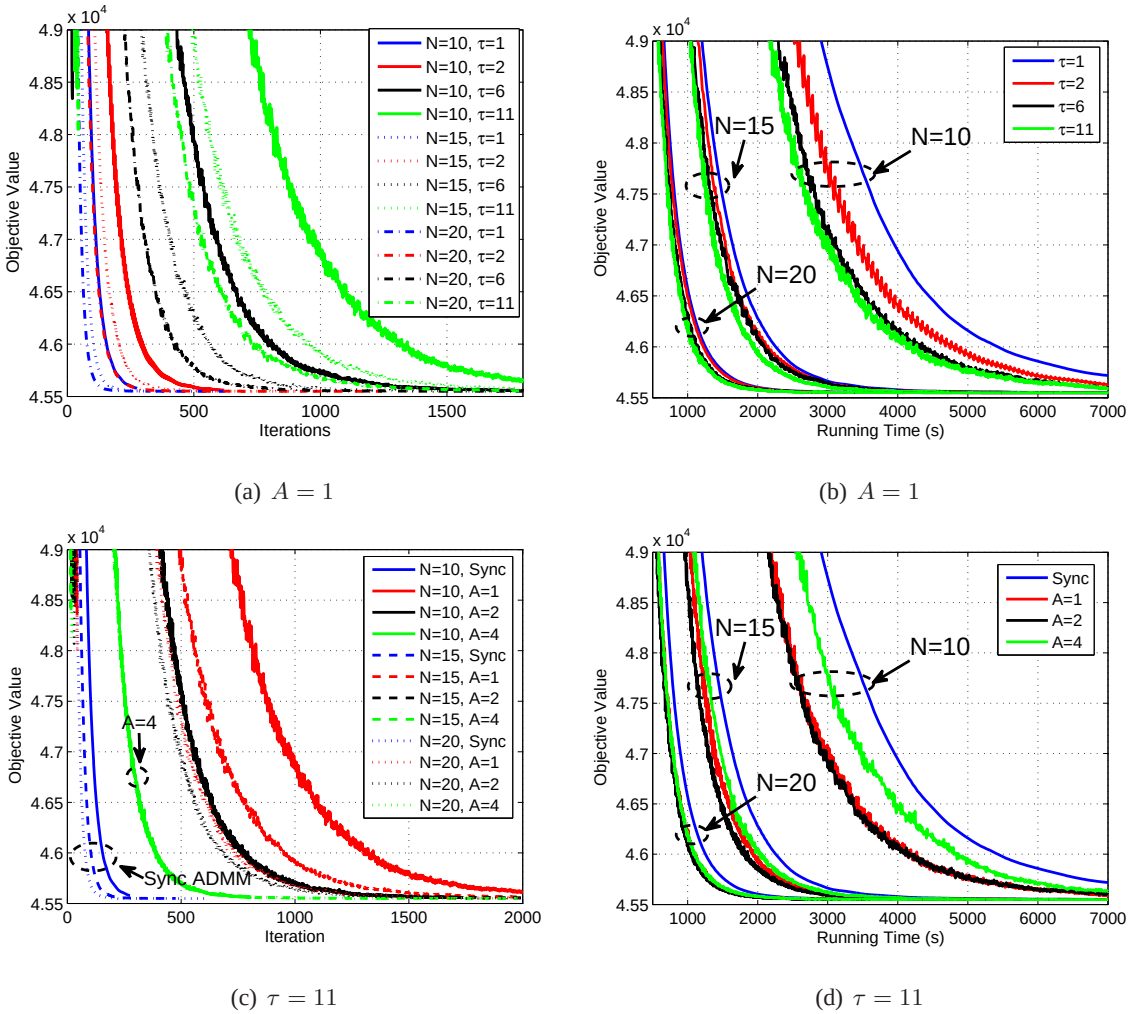
V. Numerical Results

In this section, we present some simulation results to examine the practical performance of the AD-ADMM. We consider the following LR problem

$$\min_{\boldsymbol{w}\in\mathcal{W}} \sum_{j=1}^{m}\log\left(1 + \exp(-y_j\boldsymbol{a}_j^T\boldsymbol{w})\right) \tag{49}$$

where $y_1, \ldots, y_m$ are the binary labels of the m training data, $\boldsymbol{w} \in \mathbb{R}^n$ is the regression variable and $\boldsymbol{A}_i = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m]^T \in \mathbb{R}^{m\times n}$ is the training data matrix. We used the MiniBooNE particle identification Data Set[1] which contains 130065 training samples ($m = 130065$) and the learning parameter has a size of 50 ($n = 50$). The constraint set $\mathcal{W}$ is set to $\mathcal{W} = \{\boldsymbol{w} \in \mathbb{R}^n \mid |w_i| \leq 10\ \forall i = 1, \ldots, n\}$. The AD-ADMM is implemented on an HP ProLiant BL280c G6 Linux Cluster (Itasca HPC in University of Minnesota). The n training samples are uniformly distributed to a set of N workers ($N = 10, 15, 20$). For each worker, we employed the fast iterative shrinkage thresholding algorithm (FISTA) [25] to solve the corresponding subproblem (10). The stepsize of FISTA is set to 0.0001 and the stopping condition is that the 2-norm of the gradient is less than 0.001. The penalty parameter ρ of the AD-ADMM is set to 0.01. Interestingly, while the theoretical convergence conditions in [23, Theorem 1] and Theorem 1

[1]https://archive.ics.uci.edu/ml/datasets/MiniBooNE+particle+identification



(a) $A = 1$ (b) $A = 1$

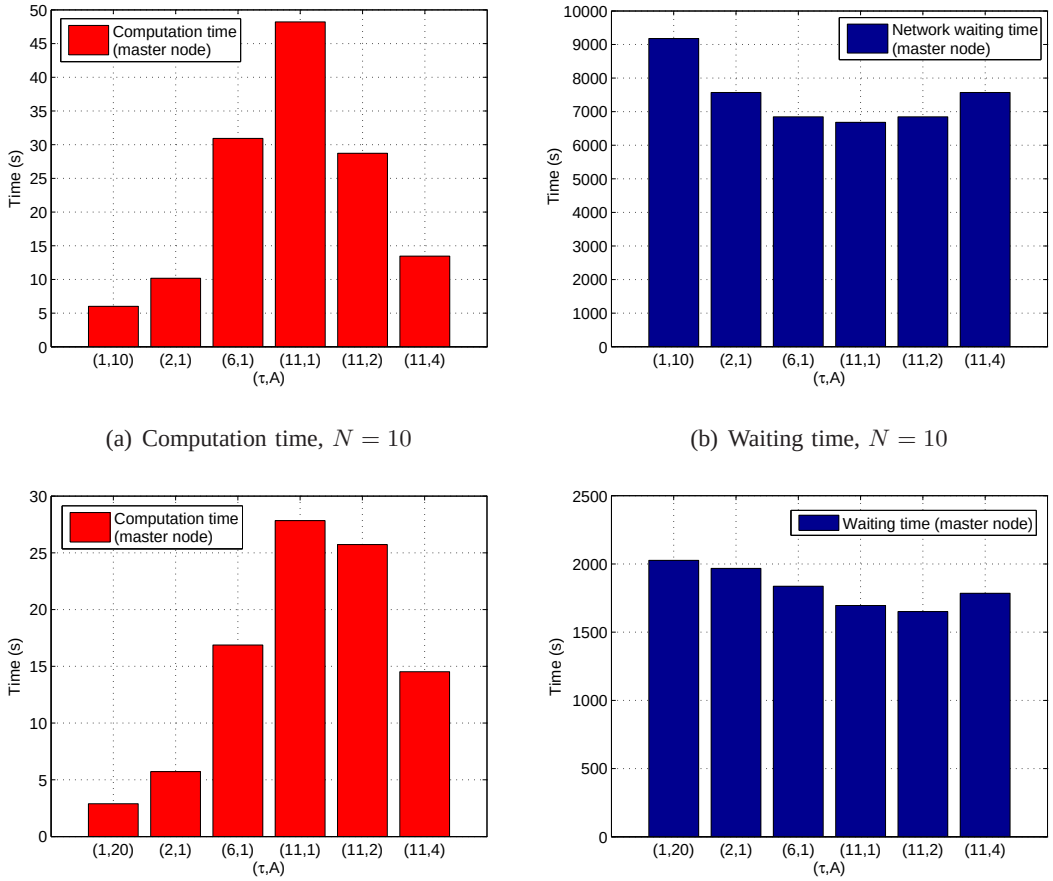
(c) $\tau = 11$ (d) $\tau = 11$

Fig. 1: Convergence curves of Algorithm 1 for solving the LR problem (49) on the Itasca computer cluster; $\theta = 0.1$, $\rho = 0.01$ and $\gamma = 0$.

all suggest that the penalty parameter γ should be large in the worst-case, we find that, for the problem instance we test here, it is also fine to set $\gamma = 0$.

Note that the asynchrony in our setting comes naturally from the heterogeneity of the computation times of computing nodes. In our experiments, analogous to [22], we further constrained the minimum size of the active set $\mathcal{A}_k$ by $|\mathcal{A}_k| \geq A$ where $A \in [1, N]$ is an integer. When $A = N$, it corresponds to the synchronous case where the master is forced to wait for all the workers at every iteration.

Figure 1(a) and Figure 1(b) respectively display the convergence curves (objective value) of the AD-ADMM versus the iteration number and the running time (second), for various values of N and τ. Here we



(a) Computation time, $N = 10$

(b) Waiting time, $N = 10$

(c) Computation time, $N = 20$

(d) Waiting time, $N = 20$

Fig. 2: The master's computation and waiting times for solving the LR problem (49) over the Itasca computer cluster.

set $A = 1$. One can observe from Figure 1(a) that, in terms of the iteration number, the convergence speed of the AD-ADMM slows down when τ increases. However, as seen from Figure 1(b), the AD-ADMM is actually faster than its synchronous counterpart ($\tau = 1$), and the running time of the AD-ADMM can be further reduced with increased τ. We also observe that, when N increases, the advantages of the AD-ADMM compared to its synchronous counterpart reduces. This is because the computation load allocated to each worker decreases with N (as n is fixed), making all the workers experience similar computation delays.

In Figure 1(c) and Figure 1(d), we present the convergence curves of AD-ADMM with different values of A. We see from Figure 1(c) that when A increases, it always requires fewer number of iterations to

achieve convergence for all choices of parameters. From Figure 1(d), however we can observe that a larger value of A is not always beneficial in reducing the running time. Specifically, one can see that for $N = 10$, the running time of AD-ADMM decreases when one increases A from 1 to 2, whereas the running time increases a lot if one increases A to 4. One can observe similar results for $N = 15$ and $N = 20$.

To look into how the values of τ and A impact on the algorithm speed, in Figure 2, we respectively plot the computation time and the waiting time of the master node for various pairs of (τ, A). The setting is the same as that in Figure 1, except that here the stopping condition of the AD-ADMM is that the objective value achieves 4.56×10^4. One can observe from these figures that, when τ increases, the computing load of the master also increases but the waiting time is significantly reduced. This explains why in Figure 1(b) the AD-ADMM requires a less running time compared with the synchronous ADMM. On the other hand, when A increases, the computation time of the master always decreases. This is because the master may take a smaller number of iterations to reach the target objective value (see Figure 1(c)) and have to spend more time waiting for slow workers. However, the overall waiting time of the master does not necessarily become larger or smaller with A. As seen from Figure 2(b) and Figure 2(d), when A increases from 1 to 2, the waiting time for $N = 10$ in Figure 2(b) increases, whereas the waiting time for $N = 20$ in Figure 2(d) decreases. However, for $A = 4$, the waiting times always become larger. Nevertheless, when comparing to the synchronous ADMM (i.e., $(\tau, A) = (1, N)$), we can see that the waiting time of the master in the AD-ADMM is always much smaller.

VI. Conclusions

In this paper, we have analytically studied the linear convergence conditions of the AD-ADMM proposed in [23]. Specifically, we have shown that for strongly convex f_i's (Assumption 3) or for f_i's with the composite form in Assumption 4, the AD-ADMM is guaranteed to converge linearly, provided that the penalty parameter ρ and the proximal parameter γ are chosen sufficiently large depending on the delay τ. When the delay τ is bounded and N is large, we have further shown that linear convergence can be achieved with zero proximal parameter (i.e.,$\gamma = 0$), and with a delay-independent ρ. The linear convergence conditions and the linear rate have been given explicitly, which relate the algorithm and network parameters with the algorithm worst-case convergence performance. The presented numerical examples have shown that in practice the AD-ADMM can effectively reduce the waiting time of the master node, and as a consequence improves the overall time efficiency of distributed optimization significantly.

APPENDIX A
BOUND OF CONSENSUS ERROR

We bound the size of the consensus error $\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{k+1}\|^2$ in the following lemma.

Lemma 6 *Under Assumption 2, it holds that*

$$\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{k+1}\|^2 \le \frac{2L^2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^{k}\|^2 + \frac{2L^2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2 + 4\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^{k}\|^2 + 4\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^{k}\|^2 + 4N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^{k}\|^2. \tag{A.1}$$

Proof: It follows from (22) and (29) that the following chain is true

$$\begin{aligned}
\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{k+1}\|^2 &= \sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\bar{k}_i+1}+\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^{k+1}\|^2 \\
&\quad+\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\widehat{k}_i+1}+\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^{k+1}\|^2 \\
&\le 2\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2+2\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2 \\
&\quad+2\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^{k+1}\|^2+2\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^{k+1}\|^2 \\
&\le \frac{2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{\lambda}_i^{k+1}-\boldsymbol{\lambda}_i^{k}\|^2+\frac{2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{\lambda}_i^{\widetilde{k}_i+1}-\boldsymbol{\lambda}_i^{\widetilde{k}_i}\|^2 \\
&\quad+2\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^{k}+\boldsymbol{x}_0^{k}-\boldsymbol{x}_0^{k+1}\|^2+2\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^{k}+\boldsymbol{x}_0^{k}-\boldsymbol{x}_0^{k+1}\|^2 \\
&\le \frac{2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{\lambda}_i^{k+1}-\boldsymbol{\lambda}_i^{k}\|^2+\frac{2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{\lambda}_i^{\widetilde{k}_i+1}-\boldsymbol{\lambda}_i^{\widetilde{k}_i}\|^2 \\
&\quad+4\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^{k}\|^2+4\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^{k}\|^2+4N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^{k}\|^2.
\end{aligned} \tag{A.2}$$

Recall from [23, Eqn. (38)] that

$$\nabla f_i(\boldsymbol{x}_i^{k+1})+\boldsymbol{\lambda}_i^{k+1}=\mathbf{0}\ \forall i\in\mathcal{V}\text{ and }\forall k. \tag{A.3}$$

By substituting (A.3) into (A.2) and by the Lipschitz continuity of ∇f_i, we obtain (A.1). ■

APPENDIX B
PROOF OF LEMMA 2

It is easy to show the following chain is true

$$\begin{aligned}\sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{N}_j}\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j_i+1}\|^2 &= \sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{N}_j}\|\sum_{q=j_i+1}^{j-1}(\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1})\|^2\\ &\le \sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{N}_j}(j-j_i-1)\sum_{q=j_i+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2\\ &\le \sum_{j=0}^{k}\eta^j\sum_{i\in\mathcal{N}_j}(\nu-1)\sum_{q=j-\nu+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2\\ &\le (\nu-1)\bar{N}\bigg(\sum_{j=0}^{k}\eta^j\sum_{q=j-\nu+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2\bigg),\end{aligned}\tag{A.4}$$

where the second inequality is owing to $j-\nu\le j_i$. To proceed, we list $\eta^j\sum_{q=j-\nu+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2$ for $j=1,\ldots,\nu,\ldots$, below

$$\begin{aligned}
&j=1, && \eta\sum_{q=2-\nu}^{0}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2=\eta\|\boldsymbol{x}_0^0-\boldsymbol{x}_0^1\|^2\\
&j=2, && \eta^2\sum_{q=3-\nu}^{1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2=\eta^2\|\boldsymbol{x}_0^0-\boldsymbol{x}_0^1\|^2+\eta^2\|\boldsymbol{x}_0^1-\boldsymbol{x}_0^2\|^2\\
&\quad\vdots && \quad\vdots\\
&j=\nu-1, && \eta^{\nu-1}\sum_{q=0}^{\nu-2}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2=\eta^{\nu-1}\|\boldsymbol{x}_0^0-\boldsymbol{x}_0^1\|^2+\eta^{\nu-1}\|\boldsymbol{x}_0^1-\boldsymbol{x}_0^2\|^2+\cdots+\eta^{\nu-1}\|\boldsymbol{x}_0^{\nu-2}-\boldsymbol{x}_0^{\nu-1}\|^2\\
&j=\nu, && \eta^{\nu}\sum_{q=1}^{\nu-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2=\eta^{\nu}\|\boldsymbol{x}_0^1-\boldsymbol{x}_0^2\|^2+\eta^{\nu}\|\boldsymbol{x}_0^2-\boldsymbol{x}_0^3\|^2+\cdots+\eta^{\nu}\|\boldsymbol{x}_0^{\nu-1}-\boldsymbol{x}_0^{\nu}\|^2
\end{aligned}\tag{A.5}$$

One can verify that each $\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j+1}\|^2$ appears no more than $\nu-1$ times in the summation term $\sum_{j=0}^{k}\eta^j\sum_{q=j-\nu+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2$ and therefore the total contribution of each $\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j+1}\|^2$ can be upper bounded by

$$(\eta^{j+1}+\eta^{j+2}+\cdots+\eta^{j+\nu-1})\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j+1}\|^2=\eta^{j+1}\bigg(\frac{\eta^{\nu-1}-1}{\eta-1}\bigg)\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j+1}\|^2.\tag{A.6}$$

This shows that

$$\sum_{j=0}^{k}\eta^j\sum_{q=j-\nu+1}^{j-1}\|\boldsymbol{x}_0^q-\boldsymbol{x}_0^{q+1}\|^2\le\sum_{j=0}^{k-1}\eta^{j+1}\bigg(\frac{\eta^{\nu-1}-1}{\eta-1}\bigg)\|\boldsymbol{x}_0^j-\boldsymbol{x}_0^{j+1}\|^2.\tag{A.7}$$

By substituting (A.7) into (A.4), we obtain (31). ■

APPENDIX C
PROOF OF LEMMA 3

By the optimality condition of (21) [24] , one has, $\forall i \in \mathcal{A}_k$ and $\forall \boldsymbol{x}_i \in \mathbb{R}^n$,

$$\begin{aligned} 0 &\geq (\nabla f_i(\boldsymbol{x}_i^{k+1}) + \boldsymbol{\lambda}_i^k + \rho(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{\bar{k}_i+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) \\ &= (\nabla f_i(\boldsymbol{x}_i^{k+1}))^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) + (\boldsymbol{\lambda}_i^{k+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i), \end{aligned} \tag{A.8}$$

where the equality is due to (22). Similarly, by the optimality condition of (28) and by (29), one has, $\forall i \in \mathcal{A}_k^c$ and $\forall \boldsymbol{x}_i \in \mathbb{R}^n$,

$$\begin{aligned} 0 &\geq (\nabla f_i(\boldsymbol{x}_i^{k+1}) + \boldsymbol{\lambda}_i^{\widetilde{k}_i} + \rho(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{\widehat{k}_i+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) \\ &= (\nabla f_i(\boldsymbol{x}_i^{k+1}))^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) + (\boldsymbol{\lambda}_i^{k+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i). \end{aligned} \tag{A.9}$$

Summing (A.8) and (A.9) for all $i \in \mathcal{V}$ gives rise to

$$\begin{aligned} &\sum_{i=1}^N (\nabla f_i(\boldsymbol{x}_i^{k+1}))^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) + \sum_{i=1}^N (\boldsymbol{\lambda}_i^{k+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i) \\ &\leq 0 \quad \forall (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) \in \mathbb{R}^{nN}. \end{aligned} \tag{A.10}$$

In addition, by the optimality condition of (23) [7, Lemma 4.1], one has, $\forall \boldsymbol{x}_0 \in \mathbb{R}^n$,

$$\begin{aligned} h(\boldsymbol{x}_0^{k+1}) - h(\boldsymbol{x}_0) &- \sum_{i=1}^N (\boldsymbol{\lambda}_i^{k+1})^T(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0) \\ &- \rho \sum_{i=1}^N (\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1})^T(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0) \\ &+ \gamma(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k)^T(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0) \leq 0. \end{aligned} \tag{A.11}$$

Denote $\boldsymbol{x}^\star \in \mathbb{R}^n$ as an optimal solution to problem (1). Let $\boldsymbol{x}_1 = \cdots = \boldsymbol{x}_N = \boldsymbol{x}_0 = \boldsymbol{x}^\star$ in (A.10) and (A.11), and combine the two equations. We obtain

$$\begin{aligned} &\sum_{i=1}^N (\nabla f_i(\boldsymbol{x}_i^{k+1}))^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star) + h(\boldsymbol{x}_0^{k+1}) - h(\boldsymbol{x}^\star) \\ &\quad + \sum_{i=1}^N (\boldsymbol{\lambda}_i^{k+1})^T(\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}) \\ &\quad - \rho \sum_{i=1}^N (\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1})^T(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star) \\ &\quad + \gamma(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k)^T(\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star) \leq 0. \end{aligned} \tag{A.12}$$

Let $\boldsymbol{y} = \boldsymbol{x}^\star$ and $\boldsymbol{x} = \boldsymbol{x}_i^{k+1}$ in (32) for all $i \in \mathcal{V}$, and apply them to (A.12). We have

$$\begin{aligned} 0 \geq & \left(\sum_{i=1}^N f_i(\boldsymbol{x}_i^{k+1}) + h(\boldsymbol{x}_0^{k+1}) - \sum_{i=1}^N f_i(\boldsymbol{x}^\star) - h(\boldsymbol{x}^\star) \right) \\ & + \sum_{i=1}^N (\boldsymbol{\lambda}_i^{k+1})^T (\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}) + \frac{\sigma^2}{2} \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star\|^2 \\ & - \rho \sum_{i=1}^N (\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1})^T (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star) \\ & + \gamma (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k)^T (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star), \end{aligned} \tag{A.13}$$

Note that, by (20),

$$\begin{aligned} & - \rho \sum_{i=1}^N (\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1})^T (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star) \\ & = -\frac{\rho}{2} \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star\|^2 + \frac{\rho}{2} \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}\|^2 \\ & \quad + \frac{\rho N}{2} \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star\|^2, \end{aligned} \tag{A.14}$$

and that

$$\begin{aligned} \gamma (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k)^T (\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star) = & \frac{\gamma}{2} \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star\|^2 \\ & - \frac{\gamma}{2} \|\boldsymbol{x}_0^k - \boldsymbol{x}^\star\|^2 + \frac{\gamma}{2} \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2. \end{aligned} \tag{A.15}$$

By substituting (A.14) and (A.15) into (A.13) and recalling $\mathcal{L}_\rho$ in (12), we obtain

$$\begin{aligned} \triangle_{k+1} \leq & \frac{\rho - \sigma^2}{2} \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star\|^2 + \frac{\gamma}{2} \|\boldsymbol{x}_0^k - \boldsymbol{x}^\star\|^2 \\ & - \frac{\gamma}{2} \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2 - \frac{\gamma + \rho N}{2} \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star\|^2. \end{aligned} \tag{A.16}$$

We bound the term $\sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star\|^2$ as

$$\begin{aligned} & \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}^\star\|^2 = \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1} + \boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star\|^2 \\ & \leq (1 + \frac{1}{\delta}) N \|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}^\star\|_2^2 + (1 + \delta) \sum_{i=1}^N \|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}\|^2 \qquad \text{(by (20))} \end{aligned}$$

$$\leq (1+\frac{1}{\delta})N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}^\star\|_2^2+(1+\delta)\bigg[\frac{2L^2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2+\frac{2L^2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2$$

$$+4\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^k\|^2+4\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^k\|^2+4N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2\bigg] \qquad \text{(by (A.1))}$$

$$\leq (1+\frac{1}{\delta})N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}^\star\|_2^2+\frac{4\delta L^2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2+\frac{4\delta L^2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2$$

$$+8\delta\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^{\bar{k}_i+1}-\boldsymbol{x}_0^k\|^2+8\delta\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^{\widehat{k}_i+1}-\boldsymbol{x}_0^k\|^2+8\delta N\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2, \tag{A.17}$$

where the last inequality is obtained by assuming $\delta>1$. Besides, we bound the term $\frac{\gamma}{2}\|\boldsymbol{x}_0^k-\boldsymbol{x}^\star\|^2$ in the RHS of (A.16) as

$$\begin{aligned}\frac{\gamma}{2}\|\boldsymbol{x}_0^k-\boldsymbol{x}^\star\|^2&=\frac{\gamma}{2}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{k+1}+\boldsymbol{x}_0^{k+1}-\boldsymbol{x}^\star\|^2\\&\leq\frac{\gamma}{2}(1+\delta)\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{k+1}\|^2+\frac{\gamma}{2}(1+\frac{1}{\delta})\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}^\star\|^2.\end{aligned} \tag{A.18}$$

By substituting (A.17) and (A.18) into (A.16), one obtains

$$\begin{aligned}\triangle_{k+1}\leq&\left(\frac{\rho N+\gamma}{2\delta}-\frac{\sigma^2N}{2}(1+\frac{1}{\delta})\right)\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}^\star\|^2\\&+\left(\frac{\gamma\delta}{2}+4(\rho-\sigma^2)N\delta\right)\|\boldsymbol{x}_0^{k+1}-\boldsymbol{x}_0^k\|^2+\frac{2(\rho-\sigma^2)\delta L^2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_i^k\|^2\\&+\frac{2(\rho-\sigma^2)\delta L^2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\widetilde{k}_i+1}-\boldsymbol{x}_i^{\widetilde{k}_i}\|^2\\&+4(\rho-\sigma^2)\delta\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\bar{k}_i+1}\|^2+4(\rho-\sigma^2)\delta\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k-\boldsymbol{x}_0^{\widehat{k}_i+1}\|^2.\end{aligned} \tag{A.19}$$

Let $\delta>1$ be large enough so that $\frac{\rho N+\gamma}{2\delta}-\frac{\sigma^2N}{2}(1+\frac{1}{\delta})\leq 0$ and assume that $\gamma\geq 8(\rho-\sigma^2)N$. Then, one obtains (33) from (A.19).

To show (34), let $\gamma=0$ in (A.19) and assume that $\delta>1$ be large enough so that $\frac{\rho}{\delta}-\sigma^2(1+\frac{1}{\delta})\leq 0$. ■

APPENDIX D
PROOF OF LEMMA 4

Since $\mathcal{X}^\star$ is a linear set, according to the Hoffman bound [26], for some constant $c>0$,

$$\begin{aligned}\text{dist}^2(\mathcal{X}^\star,\hat{\boldsymbol{x}})&=\sum_{i=1}^N\|\mathcal{P}^\star(\hat{\boldsymbol{x}})-\boldsymbol{x}_i\|^2+\|\mathcal{P}^\star(\hat{\boldsymbol{x}})-\boldsymbol{x}_0\|^2\\&\leq c\sum_{i=1}^N\|\boldsymbol{A}_i\boldsymbol{x}_i-\boldsymbol{y}_i^\star\|^2+c\sum_{i=1}^N\|\boldsymbol{x}_i-\boldsymbol{x}_0\|^2.\end{aligned} \tag{A.20}$$

In addition, it follows from the strong convexity of g_i's that

$$\begin{aligned}
&\sum_{i=1}^{N} f_i(\mathcal{P}^\star(\hat{\boldsymbol{x}})) = \sum_{i=1}^{N} g_i(\boldsymbol{A}_i\mathcal{P}^\star(\hat{\boldsymbol{x}})) \\
&\geq \sum_{i=1}^{N} g_i(\boldsymbol{A}_i\boldsymbol{x}_i) + \sum_{i=1}^{N}(\nabla g_i(\boldsymbol{A}_i\boldsymbol{x}_i))^T\boldsymbol{A}_i(\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{x}_i) + \sum_{i=1}^{N}\frac{\sigma^2}{2}\|\boldsymbol{A}_i\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{A}_i\boldsymbol{x}_i\|^2 \\
&= \sum_{i=1}^{N} f_i(\boldsymbol{x}_i) + \sum_{i=1}^{N}(\nabla f_i(\boldsymbol{x}_i))^T(\mathcal{P}^\star(\hat{\boldsymbol{x}}) - \boldsymbol{x}_i) + \sum_{i=1}^{N}\frac{\sigma^2}{2}\|\boldsymbol{y}_i^\star - \boldsymbol{A}_i\boldsymbol{x}_i\|^2. \qquad \text{(A.21)}
\end{aligned}$$

By substituting (A.20) into (A.21), one obtains (46). ■

APPENDIX E
PROOF OF LEMMA 5

By applying (46) (with $\boldsymbol{x}_i = \boldsymbol{x}_i^{k+1}\ \forall i = 0, 1, \ldots, N$) to (A.12), and following the same steps as in (A.13)-(A.16), we have

$$\begin{aligned}
\triangle_{k+1} \leq\ & \frac{\rho - \sigma^2/c}{2}\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k+1} - \mathcal{P}^\star(\hat{\boldsymbol{x}})\|^2 + \frac{\gamma}{2}\|\boldsymbol{x}_0^k - \mathcal{P}^\star(\hat{\boldsymbol{x}})\|^2 \\
&- \frac{\gamma}{2}\|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2 - \frac{\gamma + \sigma^2/c + \rho N}{2}\|\boldsymbol{x}_0^{k+1} - \mathcal{P}^\star(\hat{\boldsymbol{x}})\|^2 \\
&+ \frac{\sigma^2}{2}\sum_{i=1}^{N}\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_0^{k+1}\|^2. \qquad \text{(A.22)}
\end{aligned}$$

Recall (A.17), (A.18) (with $\boldsymbol{x}^\star$ replaced by $\mathcal{P}^\star(\hat{\boldsymbol{x}})$) and (A.1) in Lemma 6 and apply them to (A.22). One obtains

$$\begin{aligned}
\triangle_{k+1} \leq\ & \left(\frac{\rho N + \gamma}{2\delta} - \frac{N\sigma^2/c}{2}(1 + \frac{1}{\delta})\right)\|\boldsymbol{x}_0^{k+1} - \mathcal{P}^\star(\hat{\boldsymbol{x}})\|^2 \\
&+ \left(\frac{\gamma\delta}{2} + 4(\rho - \sigma^2/c)N\delta + 2\sigma^2 N\right)\|\boldsymbol{x}_0^{k+1} - \boldsymbol{x}_0^k\|^2 \\
&+ \frac{(2(\rho - \sigma^2/c)\delta + \sigma^2)L^2}{\rho^2}\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_i^{k+1} - \boldsymbol{x}_i^k\|^2 \\
&+ \frac{(2(\rho - \sigma^2/c)\delta + \sigma^2)L^2}{\rho^2}\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_i^{\tilde{k}_i+1} - \boldsymbol{x}_i^{\tilde{k}_i}\|^2 \\
&+ (4(\rho - \sigma^2/c)\delta + 2\sigma^2)\sum_{i\in\mathcal{A}_k}\|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\bar{k}_i+1}\|^2 \\
&+ (4(\rho - \sigma^2/c)\delta + 2\sigma^2)\sum_{i\in\mathcal{A}_k^c}\|\boldsymbol{x}_0^k - \boldsymbol{x}_0^{\hat{k}_i+1}\|^2. \qquad \text{(A.23)}
\end{aligned}$$

Let $\delta > 1$ be large enough so that $\frac{\rho N+\gamma}{2\delta} - \frac{N\sigma^2/c}{2}(1+\frac{1}{\delta}) \leq 0$ In addition, since $\gamma \geq 8N(\rho - \sigma^2/c) + 4N\sigma^2$ implies $\gamma \geq 8N(\rho - \sigma^2/c) + 4N\sigma^2/\delta$, (A.23) infers (33).

To obtain (48), let $\gamma = 0$ in (A.23) and assume that $\delta > 1$ be large enough so that $\frac{\rho}{\delta} - \frac{\sigma^2}{c}(1+\frac{1}{\delta}) \leq 0$.

■

REFERENCES

[1] T.-H. Chang, M. Hong, W.-C. Liao, and X. Wang, "Asynchronous distributed alternating direction method of multipliers: Algorithm and convergence analysis," submitted to *NIPS*, Montreal, Canada, Dec. 7-12, 2015.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[3] R. Tibshirani and M. Saunders, "Sparisty and smoothness via the fused lasso," *J. R. Statist. Soc. B*, vol. 67, no. 1, pp. 91–108, 2005.

[4] J. Liu, J. Chen, and J. Ye, "Large-scale sparse logistic regression," in *Proc. ACM Int. Conf. on Knowledge Discovery and Data Mining*, New York, NY, USA, June 28 - July 1, 2009, pp. 547–556.

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer-Verlag, 2001.

[6] P. Richtárik, M. Takáč, and S. D. Ahipasaoglu, "Alternating maximization: Unifying framework for 8 sparse PCA formulations and efficient parallel codes," [Online] http://arxiv.org/abs/1212.4137.

[7] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: Numerical methods*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989.

[8] F. Niu, B. Recht, C. Re, and S. J. Wright, "Hogwild!: A lock-free approach to parallelizing stochastic gradient descent," *Proc. Advances in Neural Information Processing Systems (NIPS)*, vol. 24, pp. 693-701, 2011, [Online] http://arxiv.org/abs/1106.5730.

[9] A. Agarwal and J. C. Duchi, "Distributed delayed stochastic optimization," *Proc. Advances in Neural Information Processing Systems (NIPS)*, vol. 24, pp. 873-881, 2011, [Online] http://arxiv.org/abs/1104.5525.

[10] M. Li, L. Zhou, Z. Yang, A. Li, F. Xia, D. G. Andersen, and A. Smola, "Parameter server for distributed machine learning," [Online] http://www.cs.cmu.edu/~muli/file/ps.pdf.

[11] M. Li, D. G. Andersen, and A. Smola, "Distributed delayed proximal gradient methods," [Online] http://www.cs.cmu.edu/~muli/file/ddp.pdf.

[12] J. Liu and S. J. Wright, "Asynchronous stochastic coordinate descent: Parallelism and convergence properties," *SIAM J. Optim.,*, vol. 25, no. 1, pp. 351–376, Feb. 2015.

[13] M. Razaviyayn, M. Hong, Z.-Q. Luo, and J. S. Pang, "Parallel successive convex approximation for nonsmooth nonconvex optimization," in *the Proceedings of the Neural Information Processing (NIPS)*, 2014.

[14] G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J.-S. Pang, "Decomposition by partial linearization: Parallel optimization of multi-agent systems," *IEEE Transactions on Signal Processing*, vol. 63, no. 3, pp. 641–656, 2014.

[15] B. He and X. Yuan, "On the $o(1/n)$ convergence rate of Douglas-Rachford alternating direction method," *SIAM J. Num. Anal.*, vol. 50, 2012.

[16] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," technical report; available on http://arxiv.org/pdf/1410.1390.pdf.

[17] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," Rice CAAM technical report 12-14, 2012.

[18] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1750–1761, April 2014.

[19] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 482–497, Jan. 2015.

[20] D. Jakovetić, J. M. F. Moura, and J. Xavier, "Linear convergence rate of class of distributed augmented lagrangian algorithms," to appear in *IEEE Trans. Automatic Control*.

[21] M. Hong and Z.-Q. Luo, "On the linear convergence of the alternating direction method of multipliers," available on arxiv.org.

[22] R. Zhang and J. T. Kwok, "Asynchronous distributed ADMM for consensus optimization," in *Proc. 31th ICML, , 2014.*, Beijing, China, June 21-26, 2014, pp. 1–9.

[23] T.-H. Chang, M. Hong, W.-C. Liao, and X. Wang, "Asynchronous distributed ADMM for large-scale optimization- Part I: Algorithm and convergence analysis," submitted for publication.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

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

[26] A. J. Hoffman, "On approximate solutions of systems of linear inequalities," *Journal of Research of the National Bureau of Standards,*, vol. 49, pp. 263–265, 1952.

Algorithm 1 Asynchronous Distributed ADMM for (2).

Algorithm of the Master:

Given initial variable $\boldsymbol{x}^0$ and broadcast it to the workers. Set $k=0$ and $d_1=\cdots=d_N=0$;

repeat

wait until receiving $\{\hat{\boldsymbol{x}}_i,\hat{\boldsymbol{\lambda}}_i\}_{i\in\mathcal{A}_k}$ from workers $i\in\mathcal{A}_k$ and that $d_i<\tau-1\ \forall i\in\mathcal{A}_k^c$.

update

$$\boldsymbol{x}_i^{k+1}=\begin{cases}\hat{\boldsymbol{x}}_i & \forall i\in\mathcal{A}_k\\ \boldsymbol{x}_i^k & \forall i\in\mathcal{A}_k^c\end{cases}, \tag{6}$$

$$\boldsymbol{\lambda}_i^{k+1}=\begin{cases}\hat{\boldsymbol{\lambda}}_i & \forall i\in\mathcal{A}_k\\ \boldsymbol{\lambda}_i^k & \forall i\in\mathcal{A}_k^c\end{cases}, \tag{7}$$

$$d_i=\begin{cases}0 & \forall i\in\mathcal{A}_k\\ d_i+1 & \forall i\in\mathcal{A}_k^c\end{cases}, \tag{8}$$

$$\begin{aligned}\boldsymbol{x}_0^{k+1}=&\arg\min_{\boldsymbol{x}_0\in\mathbb{R}^n}\Big\{h(\boldsymbol{x}_0)-\boldsymbol{x}_0^T\textstyle\sum_{i=1}^N\boldsymbol{\lambda}_i^{k+1}\\&+\tfrac{\rho}{2}\textstyle\sum_{i=1}^N\|\boldsymbol{x}_i^{k+1}-\boldsymbol{x}_0\|^2+\tfrac{\gamma}{2}\|\boldsymbol{x}_0-\boldsymbol{x}_0^k\|^2\Big\},\end{aligned} \tag{9}$$

broadcast $\boldsymbol{x}_0^{k+1}$ to the workers in $\mathcal{A}_k$.

set $k\leftarrow k+1$.

until a predefined stopping criterion is satisfied.

Algorithm of the ith Worker:

Given initial $\boldsymbol{\lambda}^0$ and set $k_i=0$.

repeat

wait until receiving $\hat{\boldsymbol{x}}_0$ from the master node.

update

$$\boldsymbol{x}_i^{k_i+1}=\arg\min_{\boldsymbol{x}_i\in\mathbb{R}^n}f_i(\boldsymbol{x}_i)+\boldsymbol{x}_i^T\boldsymbol{\lambda}_i^{k_i}+\tfrac{\rho}{2}\|\boldsymbol{x}_i-\hat{\boldsymbol{x}}_0\|^2, \tag{10}$$

$$\boldsymbol{\lambda}_i^{k_i+1}=\boldsymbol{\lambda}_i^{k_i}+\rho(\boldsymbol{x}_i^{k_i+1}-\hat{\boldsymbol{x}}_0). \tag{11}$$

send $(\boldsymbol{x}_i^{k_i+1},\boldsymbol{\lambda}_i^{k_i+1})$ to the master node.

set $k_i\leftarrow k_i+1$.

until a predefined stopping criterion is satisfied.