Journal of Machine Learning Research 26 (2025) 1-34 Submitted 2/25; Revised 9/25; Published 11/25

Optimal Decentralized Composite Optimization for Strongly Convex Functions

Haishan Ye YEHAISHAN@XJTU.EDU.CN
Center for Intelligent Decision-Making and Machine Learning
School of Management
Xi'an Jiaotong University

Xiangyu Chang * XIANGYUCHANG@XJTU.EDU.CN
Center for Intelligent Decision-Making and Machine Learning
School of Management
Xi'an Jiaotong University

Editor: Peter Richtarik

Abstract

This paper concentrates on decentralized composite optimization for strongly convex functions. Specifically, we first study the case where each local objective function $f_i(x)$ held by agent i is L-smooth and convex, while the regularization term $g(x)$ is μ-strongly convex. For this problem class, we propose the first decentralized algorithm that simultaneously achieves the optimal computation and communication complexities. Furthermore, we extend our algorithm to two broader scenarios. In the first extension, when each $f_i(x)$ is L-smooth and μ-strongly convex while $g(x)$ is merely convex, our algorithm continues to attain the optimal complexities. In the second extension, we show that under time-varying communication networks, our algorithm matches the lower bounds on decentralized optimization established in Kovalev et al. (2021). Finally, extensive experiments validate both the computational and communication efficiency of the proposed algorithms.

Keywords: Decentralized optimization, Composite optimization, Accelerated proximal gradient method

1. Introduction

In contemporary settings, datasets of considerable size are typically distributed across multiple storage systems. Centralizing these extensive datasets proves to be impractical, primarily due to constraints in communication bandwidth and concerns regarding data privacy (Lian et al., 2017; Song et al., 2023). Addressing these bottlenecks in numerous practical applications, the concept of distributed optimization has surfaced as a viable approach, particularly within the realms of large-scale machine learning, signal processing, control, and optimization (Sayed et al., 2014; Shi et al., 2015b; Alghunaim et al., 2019; Li and Lin, 2020).

This paper considers the composite optimization problem formulated as follows:

$$F(x) = f(x) + g(x), \text{ with } f(x) = \frac{1}{m}\sum_{i=1}^{m} f_i(x), \tag{1}$$

*. Corresponding author.

Methods	Computation	Communication	Composite?
EXTRA (Shi et al., 2015b)	$\mathcal{O}\left(\frac{L}{\mu(1-\lambda_2(W))}\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\frac{L}{\mu(1-\lambda_2(W))}\log\left(\frac{1}{\epsilon}\right)\right)$	No
OPAC (Kovalev et al., 2020)	$\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W)}}\log\frac{1}{\epsilon}\right)$	No
OGT (Song et al., 2023)	$\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W)}}\log\frac{1}{\epsilon}\right)$	No
NIDS[1] Li et al. (2019)	$\mathcal{O}\left(\left(\frac{L}{\mu}+\frac{1}{1-\lambda_2(W)}\right)\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\left(\frac{L}{\mu}+\frac{1}{1-\lambda_2(W)}\right)\log\left(\frac{1}{\epsilon}\right)\right)$	**Yes**
D2P2 (Alghunaim et al., 2019)	$\mathcal{O}\left(\frac{L}{\mu(1-\lambda_2(W))}\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\frac{L}{\mu(1-\lambda_2(W))}\log\left(\frac{1}{\epsilon}\right)\right)$	**Yes**
ProxMudag (Ye et al., 2023)	$\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W)}}\log\frac{L}{\mu}\log\left(\frac{1}{\epsilon}\right)\right)$	**Yes**
ODAPG (this paper)	$\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\epsilon}\right)\right)$	$\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W)}}\log\left(\frac{1}{\epsilon}\right)\right)$	**Yes**
Lower Bound (Scaman et al., 2017)	$\Omega\left(\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\epsilon}\right)\right)$	$\Omega\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W)}}\log\left(\frac{1}{\epsilon}\right)\right)$	–

Table 1: Complexity comparisons between our algorithm and existing works for strongly (composite) convex problems.

where each $f_i(x)$ is L-smooth and convex, and the regularization term $g(x)$ is μ-strongly convex but may be non-differentiable with $0 < \mu \leq L$. The elastic-norm serves as a typical example of $g(x)$ (Zou and Hastie, 2005). Instances of problems following the structure (1) are prevalent in various domains, such as machine learning (Wu et al., 2009), model fitting (Boyd et al., 2011), and economic dispatch in power systems (Dominguez-Garcia et al., 2012). Within the *distributed composite optimization* framework, the objective function $F(x)$ is an aggregation of m local functions $f_i(x)$, each associated with one of the m agents. The agents are interconnected through an *undirected network (static or time-varying)*, with each agent having access only to its local function and communication capabilities with its neighbors. The collective goal of these agents is to collaboratively resolve the composite optimization problem as delineated in (1) distributedly.

A multitude of decentralized algorithms have been proposed in diverse contexts (smooth decentralized and decentralized composite optimization), notably those employing the primal-dual or gradient-tracking methods, as exemplified by the studies of Shi et al. (2015a); Qu and Li (2017, 2019); Di Lorenzo and Scutari (2016); Alghunaim et al. (2019); Li and Lin (2021); Kovalev et al. (2020); Xu et al. (2021); Song et al. (2023); Ye et al. (2023); Scaman et al. (2017); Maros and Jaldén (2018); Nedic et al. (2017); Pu et al. (2020). The primary objective of these algorithms is to optimize either the convergence rate or the communication complexity. Consequently, an open question persists: *Can one devise a practical decentralized algorithm that simultaneously achieves optimal computation and communication complexities in composite decentralized optimization with the formulation* (1)*?* In this paper, we endeavor to address this problem.

1. Li et al. (2019) only gave a sublinear convergence rate for `NIDS` when $g(x)$ is convex, the linear convergence rate is proved in works (Alghunaim et al., 2020; Xu et al., 2021).

1.1 Literature Review

This part provides a concise review of decentralized algorithms in the realm of composite decentralized optimization, with a specific emphasis on the differentiability of $g(x)$.

In scenarios where $g(x)$ is differentiable, various algorithms have been documented to attain a linear convergence rate, as evidenced by studies such as Shi et al. (2015b); Mokhtari et al. (2016); Yuan et al. (2019); Li et al. (2019); Qu and Li (2017, 2019); Di Lorenzo and Scutari (2016); Li and Lin (2021); Kovalev et al. (2020); Song et al. (2023); Ye et al. (2023); Scaman et al. (2017). These algorithms predominantly employ either the primal-dual method or the gradient-tracking method. Among these algorithms, Shi et al. (2015b) proposes the first decentralized optimization algorithm that can achieve the linear convergence rate without increasing communication cost iteratively, which depends on the target precision ϵ. Scaman et al. (2017) proposes the first communication optimal algorithm, that is, its communication complexity is $\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\log\frac{1}{\epsilon}\right)$ to find an ϵ-precision solution, where W is the weight matrix related to the network topology (refer to Assumption 3) and $\lambda_2(W)$ is the second largest eigenvalue of W. Kovalev et al. (2020) provides the first algorithm that can achieve both computation and communication complexities, that is, its computation complexity is $\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$. However, the method of Kovalev et al. (2020) requires multi-consensus, that is, an inner communication loop is required, to achieve the optimal complexities. Most recently, Song et al. (2023) introduces an optimal decentralized algorithm wherein each agent communicates with its neighbors only once per iteration.

Beyond these deterministic methods, several works have recently explored randomized or variance-reduced approaches for decentralized optimization when $g(x)$ is differentiable. For instance, Hendrikx et al. (2021) proposes the ADFS algorithm, an accelerated decentralized stochastic method for finite-sum problems. ADFS combines local stochastic proximal updates with decentralized communication and is shown to be optimal with respect to both computation and communication complexities in the distributed finite-sum setting. More recently, Li et al. (2020) extends the widely used EXTRA and DIGing algorithms by incorporating variance reduction techniques, resulting in VR-EXTRA and VR-DIGing, as well as their accelerated variants. These methods achieve the same stochastic gradient computation complexities as the best single-machine variance-reduced methods, while simultaneously attaining optimal or near-optimal communication complexities in the decentralized setting.

In addition, when $g(x)$ is differentiable and the communication network is time-varying, several algorithms have been proposed to guarantee linear convergence. For instance, Nedic et al. (2017) introduces the DIGing and Push-DIGing algorithms, which combine distributed inexact gradients with gradient tracking to achieve linear convergence over both undirected and directed time-varying graphs. Building upon this, Maros and Jaldén (2018) propose the PANDA algorithm, a dual ascent-based method that converges linearly while requiring fewer communication exchanges per iteration compared to DIGing. Furthermore, Pu et al. (2020) develops the Push-Pull and Gossip Push-Pull algorithms, which achieve linear convergence for strongly convex and smooth functions in both synchronous and asynchronous time-varying directed networks. In addition, variance-reduced methods have recently been extended to the time-varying network setting. Metelev et al. (2024) study decentralized finite-sum optimization where each local function has a finite-sum structure and the communication network is time-varying. They propose ADOM+VR for strongly convex prob-

lems and GT-PAGE for nonconvex problems, representing the first decentralized variance-reduction methods for time-varying networks. Moreover, they establish lower bounds on both communication and stochastic oracle complexities, showing that the proposed algorithms are near-optimal under certain conditions. More recently, Kovalev et al. (2021) establishes the first lower bounds for decentralized optimization over time-varying networks and proposes the optimal ADOM and ADOM+ algorithms, which attain these bounds under primal and dual oracle settings, respectively. These studies collectively demonstrate that linear convergence with communication efficiency can be achieved even in the more challenging time-varying network setting.

A substantial body of research also exists in instances where $g(x)$ is non-differentiable. Numerous algorithms employing gradient tracking have been adapted to decentralized composite optimization challenges involving a non-smooth regularization term, notably `PG-EXTRA` (Shi et al., 2015a) and `NIDS` (Li et al., 2019). However, the presence of the non-smooth term limits these algorithms' theoretical convergence rates to sub-linear. Recently, Alghunaim et al. (2019) introduces a primal-dual algorithm capable of achieving a linear convergence rate. Concurrently, Sun et al. (2022) proposes `SONATA`, a gradient tracking-based method, also achieving a linear convergence rate. Kovalev et al. (2022) propose algorithms for decentralized stochastic variational inequalities and achieve a communication complexity $\mathcal{O}\left(\sqrt{\frac{1}{1-\lambda_2(W)}}\frac{L}{\mu}\log\frac{1}{\epsilon}\right)$. Following Alghunaim et al. (2019), a unified framework is developed for analyzing a wide array of primal-dual and gradient tracking-based algorithms, demonstrating their capability to achieve linear convergence rates even with a nonsmooth regularization term, as seen in the extended applications of `EXTRA` (`PG-EXTRA`) (Shi et al., 2015b), and `NIDS` (Li et al., 2019).

Despite extensive studies in the literature, the convergence rates of these previous algorithms do not match the optimal convergence rate. The communication complexities of algorithms within the framework proposed by Xu et al. (2021) and Alghunaim et al. (2020) are not yet optimal. While various optimal algorithms have been proposed for scenarios with a smooth regularization term $g(x)$, the development of an optimal algorithm for decentralized composite optimization in general remains an unresolved challenge. According to our current understanding, `ProxMudag` (Ye et al., 2023) stands as the most advanced among proposed algorithms for decentralized composite optimization applicable to strongly convex functions. `ProxMudag` attains optimal computational complexity and approaches optimal communication complexity, represented as $\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\log\frac{L}{\mu}\log\frac{1}{\epsilon}\right)$, albeit with an additional $\log\frac{L}{\mu}$ factor relative to the ideal communication complexity. Moreover, for the more challenging scenario where the network is time-varying and the regularization term $g(x)$ is non-differentiable, no existing work has yet addressed this problem. In particular, whether one can design decentralized algorithms under such settings that achieve the lower bounds on computation and communication complexities established in Kovalev et al. (2021) remains a problem.

1.2 Contributions

In this paper, we endeavor to bridge the theoretical divide between smooth decentralized optimization and decentralized composite optimization. The main contributions of this paper can be summarized as follows:

- **Algorithmic design.** We propose the `ODAPG` algorithm for decentralized composite optimization. The algorithm is built upon three key components: accelerated proximal gradient descent, gradient tracking, and the FastMix technique. By combining these elements, `ODAPG` inherits multiple desirable properties, including a rapid convergence rate from accelerated proximal gradient descent (Nesterov, 2003), variance reduction from gradient tracking (Nguyen et al., 2017), and improved communication efficiency from FastMix (Liu and Morse, 2011).

- **Optimal computation and communication complexities.** We establish that `ODAPG` achieves the optimal computation and communication complexities (see Table 1, which is included for clear comparative analysis, juxtaposing these algorithms with their state-of-the-art counterparts). The critical factor underlying this result lies in its novel acceleration scheme. Unlike `ProxMudag` (Ye et al., 2023), which employs the classical accelerated proximal gradient scheme (Beck and Teboulle, 2009; Nesterov, 2013) and leads to consensus errors bounded in the Frobenius norm—thereby incurring an additional $\log \frac{L}{\mu}$ factor even for unaccelerated methods (Sun et al., 2022; Ye et al., 2023)—our `ODAPG` algorithm leverages the recent acceleration scheme introduced by Driggs et al. (2022). This scheme bounds consensus errors in terms of extended Bregman distances (see Section 4.2), which enables us to eliminate the extra $\log \frac{L}{\mu}$ factor and thereby achieve the optimal rates.

- **Extensions to broader settings.** We further extend `ODAPG` to two important settings. First, when each $f_i(x)$ is L-smooth and μ-strongly convex while $g(x)$ is merely convex, our algorithm continues to attain the optimal computation and communication complexities. Second, in the case of time-varying communication networks, we show that our extended algorithm matches the lower bounds on decentralized optimization recently established in Kovalev et al. (2021).

2. Notation and Assumptions

Throughout this paper, we denote $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ and $\mathbf{s}$ are $m \times d$ matrices whose i-th rows $\mathbf{x}^{(i)}$, $\mathbf{y}^{(i)}$, $\mathbf{z}^{(i)}$ and $\mathbf{s}^{(i)}$ are their local copies for the i-th agent, respectively. Accordingly, we define the averaging variables

$$\begin{aligned} \overline{x} :=& \frac{1}{m}\sum_{i=1}^{m}\mathbf{x}^{(i)} = \frac{1}{m}\mathbf{1}^\top\mathbf{x} \in \mathbb{R}^{1\times d}, \quad \overline{y} := \frac{1}{m}\sum_{i=1}^{m}\mathbf{y}^{(i)} \\ \overline{z} :=& \frac{1}{m}\sum_{i=1}^{m}\mathbf{z}^{(i)}, \quad \overline{s} := \frac{1}{m}\sum_{i=1}^{m}\mathbf{s}^{(i)}, \end{aligned} \tag{2}$$

where $\mathbf{1}$ denotes the vector with all entries equal to 1 of dimension m. Now we introduce the projection matrix

$$\mathbf{\Pi} = \mathbf{I}_m - \frac{\mathbf{1}\mathbf{1}^\top}{m}. \tag{3}$$

Using the projection matrix $\mathbf{\Pi}$, we can represent that

$$\|\mathbf{x} - \mathbf{1}\overline{x}\| = \left\|\mathbf{x} - \frac{\mathbf{1}\mathbf{1}^\top}{m}\mathbf{x}\right\| = \|\mathbf{\Pi}\mathbf{x}\|.$$

To write the proposed algorithm in a compact form, we introduce an aggregate objective function $\widetilde{f}(\mathbf{x})$ defined as follows,

$$\widetilde{f}(\mathbf{x}) := \sum_{i=1}^{m} f_i(\mathbf{x}^{(i)}). \tag{4}$$

Accordingly, its aggregate gradient is

$$\nabla\widetilde{f}(\mathbf{x}) = [\nabla f_1(\mathbf{x}^{(1)}); \nabla f_2(\mathbf{x}^{(2)}); \ldots; \nabla f_m(\mathbf{x}^{(m)})] \in \mathbb{R}^{m\times d}.$$

It holds that the i-th row of $\nabla\widetilde{f}(\mathbf{x})$ equals to $\nabla f_i(\mathbf{x}^{(i)})$.

Throughout this paper, we use $\|\cdot\|$ to denote the "Frobenius" norm. That is, for a matrix $\mathbf{x} \in \mathbb{R}^{m\times d}$, it holds that

$$\|\mathbf{x}\|^2 = \sum_{i=1,j=1}^{m,d} \left(\mathbf{x}^{(i,j)}\right)^2,$$

where $\mathbf{x}^{(i,j)}$ denotes the entry of $\mathbf{x}$ in the i-th row and j-th column. Furthermore, we use $\|\mathbf{x}\|_2$ to denote the spectral norm which is the largest singular value of $\mathbf{x}$. For vectors $x, y \in \mathbb{R}^d$, we use $\langle x, y\rangle$ to denote the standard inner product of x and y.

Let us define the aggregated proximal operator with respect to $g(\cdot)$ as for $\mathbf{x} \in \mathbb{R}^{m\times d}$,

$$\mathrm{prox}_{\gamma g}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^{m\times d}} \left(\gamma g(\mathbf{w}^{(i)}) + \frac{1}{2}\left\|\mathbf{w}^{(i)} - \mathbf{x}^{(i)}\right\|^2\right).$$

Thus, the i-th row of $\mathrm{prox}_{\gamma g}(\mathbf{x})$ equals to $\mathrm{prox}_{\gamma g}(\mathbf{x}^{(i)})$.

In addition, we make the following assumptions for each local objective function and regularization term in (1).

Assumption 1 *Each $f_i : \mathbb{R}^d \to \mathbb{R}$ is convex and L-smooth, i.e., for any $x, y \in \mathbb{R}^d$, it holds that*

$$\begin{aligned} f_i(y) \geq& f_i(x) + \langle \nabla f_i(x), y - x\rangle, \\ f_i(y) \leq& f_i(x) + \langle \nabla f_i(x), y - x\rangle + \frac{L}{2}\|x - y\|^2. \end{aligned}$$

Assumption 2 *The function $g : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex, i.e., for any $x, y \in \mathbb{R}^d$, it holds that*

$$g(y) \geq g(x) + \langle \partial g(x), y - x\rangle + \frac{\mu}{2}\|x - y\|^2, \quad \textit{where} \quad \partial g(x) \textit{ is the subgradient at } x.$$

Algorithm 1 FastMix

1: **Input:** $\mathbf{x}_0$, K, W and η_w.
2: $\mathbf{x}_{-1} = \mathbf{x}_0$;
3: **for** $k = 0, \ldots, K$ **do**
4: $\quad \mathbf{x}_{k+1} = (1 + \eta_w)W\mathbf{x}_k - \eta_w \mathbf{x}_{k-1}$;
5: **end for**
6: **Output:** $\mathbf{x}_K$.

Using the $f(\cdot)$, $f_i(\cdot)$, and aggregated variable $\mathbf{x}$, we define the extended Bregman distance between $y \in \mathbb{R}^d$ and $\mathbf{x} \in \mathbb{R}^{m\times d}$ as follows:

$$D_f(y, \mathbf{x}) := \frac{1}{m}\sum_{i=1}^{m}\left(f_i(y) - f_i(\mathbf{x}^{(i)}) - \left\langle \nabla f_i(\mathbf{x}^{(i)}), y - \mathbf{x}^{(i)} \right\rangle\right). \tag{5}$$

Because of the convexity of $f_i(\cdot)$, $D_f(y, \mathbf{x})$ is non-negative for any $y \in \mathbb{R}^d$ and $\mathbf{x} \in \mathbb{R}^{m\times d}$.

For the topology of the network, let W be the weight matrix associated with the network, indicating how agents are connected to each other. The weight matrix W satisfies the following assumption.

Assumption 3 *The weight matrix W is symmetric positive semi-definite with $W_{i,j} \neq 0$ if and only if agents i and j are connected or $i = j$. It also satisfies that $\mathbf{0} \preceq W \preceq I$, $W\mathbf{1} = \mathbf{1}$, $\mathrm{null}(I - W) = \mathrm{span}(\mathbf{1})$.*

If the weight matrix W satisfies the above assumption, then one can achieve the effect of averaging local $\mathbf{x}^{(i)}$ on different agents by using $W\mathbf{x}$ for iterations. Instead of directly multiplying W, Liu and Morse (2011) proposed a more efficient way to achieve averaging described in Algorithm 1, which has the following important proposition.

Proposition 1 (Ye et al. (2023)) *Let $\mathbf{x}_K$ be the output of Algorithm 1 with $\eta_w = 1/(1 + \sqrt{1 - \lambda_2^2(W)})$ and we denote $\bar{x} = \frac{1}{m}\mathbf{1}^\top \mathbf{x}^0$. Then, it holds that*

$$\bar{x} = \frac{1}{m}\mathbf{1}^\top \mathbf{x}_K \quad \textit{and} \quad \|\mathbf{x}_K - \mathbf{1}\bar{x}\| \leq \sqrt{14}\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right)\sqrt{1 - \lambda_2(W)}\right)^K \|\mathbf{x}_0 - \mathbf{1}\bar{x}\|,$$

where $\lambda_2(W)$ is the second largest eigenvalue of W.

This proposition will be utilized in the convergence analysis of our algorithm, especially used to bind the consensus errors in Section 4.2.

3. Optimal Decentralized Accelerated Proximal Gradient Descent

This section commences with a detailed description of the algorithm. Subsequently, the primary findings of this paper are presented. Lastly, the paper extends the algorithm to address scenarios where $f_i(x)$ is L-smooth and μ-strongly convex while $g(x)$ is convex yet potentially non-differentiable.

Algorithm 2 Optimal Decentralized Accelerated Proximal Gradient Descent (`ODAPG`)

Input: x_0, mixing matrix W, initial step size γ, τ.
Initialization: Set $\mathbf{z}_0 = \mathbf{y}_0 = \mathbf{x}_0 = \mathbf{1}x_0$, $\mathbf{s}_0^{(i)} = \nabla f_i(\mathbf{x}_0^{(i)})$, in parallel for $i \in [m]$.
for $t = 1, \ldots, T$ **do**
  Compute $\mathbf{x}_{t+1} = \tau \mathbf{z}_t + (1-\tau)\mathbf{y}_t$;
  Compute the local gradients $\nabla f_i(\mathbf{x}_{t+1}^{(i)})$ in parallel for $i \in [m]$ to form the gradient $\nabla \widetilde{f}(\mathbf{x}_{t+1})$;
  $\mathbf{s}_{t+1} = \text{FastMix}(\mathbf{s}_t + \nabla \widetilde{f}(\mathbf{x}_{t+1}) - \nabla \widetilde{f}(\mathbf{x}_t), K)$;
  $\mathbf{z}_{t+1} = \text{FastMix}\Big(\text{prox}_{\gamma g}\left(\mathbf{z}_t - \gamma \mathbf{s}_{t+1}\right), K\Big)$;
  $\mathbf{y}_{t+1} = \text{FastMix}\left(\tau \mathbf{z}_{t+1} + (1-\tau)\mathbf{y}_t, K\right)$.
end for

3.1 Algorithm Description

To more efficiently address the distributed composite optimization problem as delineated in (1), we introduce a principal algorithm named Optimal Decentralized Accelerated Proximal Gradient Descent (`ODAPG`), detailed in Algorithm 2. Our algorithm synergizes the methodologies of Nesterov's accelerated gradient descent and gradient tracking. The main algorithmic procedure is listed as follows:

$$\mathbf{x}_{t+1} = \tau \mathbf{z}_t + (1-\tau)\mathbf{y}_t, \tag{6}$$

$$\mathbf{s}_{t+1} = \text{FastMix}\big(\mathbf{s}_t + \nabla \widetilde{f}(\mathbf{x}_{t+1}) - \nabla \widetilde{f}(\mathbf{x}_t), K\big), \tag{7}$$

$$\mathbf{z}_{t+1} = \text{FastMix}\left(\text{prox}_{\gamma g}(\mathbf{z}_t - \gamma \mathbf{s}_{t+1}), K\right), \tag{8}$$

$$\mathbf{y}_{t+1} = \text{FastMix}\left(\tau \mathbf{z}_{t+1} + (1-\tau)\mathbf{y}_t, K\right), \tag{9}$$

where K is the iteration number for the "FastMix" operator.

The trio of "accelerated proximal gradient descent", "gradient tracking," and "FastMix" constitutes the pivotal components that enable our algorithm to attain optimal communication and computation complexities. The "accelerated proximal gradient descent" component imbues our algorithm with the potential to achieve a rapid convergence rate. "Gradient tracking" can be conceptualized as a form of variance reduction. Its update rule parallels that of a significant variance reduction method known as SARAH (Nguyen et al., 2017). Consequently, the "gradient tracking" technique proves highly effective in reducing communication complexity and is extensively employed in decentralized optimization. For further reduction in communication complexity, "FastMix" is the key element. Without the "FastMix", the communication complexity depends on $(1-\lambda_2(W))^{-1}$. In contrast, the "FastMix" helps to achieve a communication complexity depending on $(1-\lambda_2(W))^{-1/2}$.

3.2 Main Results

This section offers an in-depth examination of the communication and computation complexities associated with `ODAPG`. Our primary emphasis is on the synchronized setting, signifying that the computation complexity is contingent upon the number of gradient calls, while the communication complexity hinges on the frequency of local communication rounds.

Theorem 2 *Let the objective function $F(x)$ be of the form* (1) *and Assumption 1-3 hold. Letting us set the step size $\gamma = \frac{1}{20\sqrt{L\mu}}$, $\tau = \mu\gamma$, and $K = \frac{11}{\sqrt{1-\lambda_2(W)}}$ in Algorithm 2, letting*

x^* denote the minimum of $F(x)$, then it holds that

$$\begin{aligned}\|\mathbf{z}_T - \mathbf{1}x^*\|^2 \leq & \left(1 - \frac{1}{40}\cdot\sqrt{\frac{\mu}{L}}\right)^T \cdot \left(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2\right. \\ & \left. + \frac{20^2 L}{2\mu}\cdot\left(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\,\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\,\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot L^2}\,\|\mathbf{\Pi}\mathbf{s}_0\|^2\right)\right).\end{aligned} \tag{10}$$

The above theorem shows that our algorithm converges with a rate $1-\mathcal{O}\left(\sqrt{\frac{\mu}{L}}\right)$, which is the same as the one of accelerated proximal gradient descent (Nesterov, 2003). Furthermore, since it holds that $\left\|\mathbf{z}_t^{(i)} - x^*\right\|^2 \leq \|\mathbf{z}_t - \mathbf{1}x^*\|^2$, Eq. (14) shows that $\mathbf{z}_t^{(i)}$ in i-th agent will converge to the optimum.

Next, by the convergence rate of our algorithm shown in Theorem 2, we will give our algorithm's computation cost and communication complexities in the following corollary.

Corollary 3 *Let the objective function satisfy the property described in Theorem 2. To find an ϵ-suboptimal solution, the computation and communication complexities of Algorithm 2 for each agent are*

$$T = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right), \quad \textit{and} \quad C = \mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\log\frac{1}{\epsilon}\right). \tag{11}$$

Proof By Theorem 2, we can obtain that

$$\begin{aligned}\|\mathbf{z}_T - \mathbf{1}x^*\|^2 \leq & \left(1 - \frac{1}{40}\cdot\sqrt{\frac{\mu}{L}}\right)^T \cdot \left(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2\right. \\ & \left. + \frac{20^2 L}{2\mu}\cdot\left(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\,\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\,\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot L^2}\,\|\mathbf{\Pi}\mathbf{s}_0\|^2\right)\right) \\ \leq & \exp\left(-\frac{1}{40}\cdot\sqrt{\frac{\mu}{L}}T\right)\cdot\left(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2\right. \\ & \left. + \frac{20^2 L}{2\mu}\cdot\left(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\,\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\,\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot L^2}\,\|\mathbf{\Pi}\mathbf{s}_0\|^2\right)\right).\end{aligned}$$

Thus, to achieve that $\left\|\mathbf{z}_t^{(i)} - x^*\right\|^2 \leq \|\mathbf{z}_T - \mathbf{1}x^*\|^2 \leq \epsilon$, T is required to be

$$\begin{aligned}T = & 40\sqrt{\frac{L}{\mu}}\left(\log\frac{1}{\epsilon} + \log\Big(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2\right. \\ & \left. + \frac{20^2 L}{2\mu}\cdot\Big(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\,\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\,\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot L^2}\,\|\mathbf{\Pi}\mathbf{s}_0\|^2\Big)\Big)\right) \\ = & \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right).\end{aligned}$$

As shown in Algorithm 2, each agent computes its local gradient $\nabla f_i(\mathbf{x}_{t+1}^{(i)})$ for each iteration. Thus, the computation complexity of our algorithm is equal to T.

Our algorithm takes three "FastMix" steps in which each agent communicates K times with its neighbors for each iteration. The communication complexity of our algorithm is

$$C = 3TK = T \cdot \frac{33}{\sqrt{1-\lambda_2(W)}} = \mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\log\frac{1}{\epsilon}\right).$$

■

Remark 4 *Eq.* (11) *demonstrates that our algorithm attains a computational complexity of* $T = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$*, aligning with the computational complexity inherent in accelerated proximal gradient descent. Consequently, our algorithm achieves the optimal computational complexity, as delineated in Nesterov (2003). Moreover, the communication complexity of our algorithm corresponds with the lower bound for decentralized optimization over strongly convex functions, as outlined in Scaman et al. (2017). Hence, our algorithm realizes the optimal benchmarks in both computational and communication complexities.*

3.3 Extension to solely convex $g(x)$

Corollary 3 establishes that our algorithm attains the optimal computational and communication complexities in scenarios where $f_i(x)$ exhibits convexity and L-smoothness, whereas $g(x)$ demonstrates μ-strong convexity. Subsequently, we demonstrate that our algorithm is adaptable to scenarios wherein $f_i(x)$ is μ-strongly convex and L-smooth, while $g(x)$ is solely convex. Simultaneously, our algorithm maintains its capability to achieve the optimal computational and communication complexities under these conditions.

The main idea of the extension of our algorithm relies on the fact that $\widehat{f}_i(x) = f_i(x) - \frac{\mu}{2}\|x\|^2$ is $(L-\mu)$-smooth and convex, $\widehat{g}(x) = g(x) + \frac{\mu}{2}\|x\|^2$ is μ-strongly convex if $f_i(x)$ is μ-strongly convex and L-smooth while $g(x)$ is only convex. Then, we can use Algorithm 2 with a step size $\widehat{\gamma} = \frac{1}{20\sqrt{(L-\mu)\mu}}$ and $\widehat{\tau} = \mu\widehat{\gamma}$ to solve the problem represented as follow:

$$F(x) = \frac{1}{m}\sum_{i=1}^{m}\widehat{f}_i(x) + \widehat{g}(x).$$

Instead, the gradient computation in Step 5 of Algorithm 2 becomes

$$\nabla\widehat{f}(\mathbf{x}_t) = \left[\nabla f_1(\mathbf{x}_t^{(1)}); \nabla f_2(\mathbf{x}_t^{(2)}); \ldots; \nabla f_m(\mathbf{x}_t^{(m)})\right] - \mu\left[\mathbf{x}_t^{(1)}; \mathbf{x}_t^{(2)}; \ldots; \mathbf{x}_t^{(m)}\right],$$

where

$$\widehat{f}(\mathbf{x}) := \widetilde{f}(\mathbf{x}) - \frac{\mu}{2}\left(\|\mathbf{x}^{(1)}\|^2 + \|\mathbf{x}^{(2)}\|^2 + \cdots + \|\mathbf{x}^{(m)}\|^2\right).$$

Algorithm 3 ODAPG for $f_i(x)$ being L-smooth and μ-strongly convex, and $g(x)$ is convex.

Input: x_0, mixing matrix W, initial step size $\widehat{\gamma}$, $\widehat{\tau}$.
Initialization: Set $\mathbf{z}_0 = \mathbf{y}_0 = \mathbf{x}_0 = \mathbf{1}x_0$, $\mathbf{s}_0^{(i)} = \nabla f_i(\mathbf{x}_0^{(i)}) - \mu x_0$, in parallel for $i \in [m]$.
for $t = 1, \ldots, T$ **do**
  Compute $\mathbf{x}_{t+1} = \widehat{\tau}\mathbf{z}_t + (1-\widehat{\tau})\mathbf{y}_t$;
  Compute the local gradients $\nabla \widehat{f}_i(\mathbf{x}_{t+1}^{(i)}) = \nabla f_i(\mathbf{x}_{t+1}^{(i)}) - \mu \mathbf{x}_{t+1}^{(i)}$ in parallel for $i \in [m]$ to form the gradient $\nabla \widehat{f}(\mathbf{x}_{t+1}) = \left[\nabla \widehat{f}_i(\mathbf{x}_{t+1}^{(1)}); \nabla \widehat{f}_i(\mathbf{x}_{t+1}^{(2)}); \ldots; \nabla \widehat{f}_i(\mathbf{x}_{t+1}^{(m)})\right]$;
  $\mathbf{s}_{t+1} = \text{FastMix}(\mathbf{s}_t + \nabla \widehat{f}(\mathbf{x}_{t+1}) - \nabla \widehat{f}(\mathbf{x}_t), K)$;
  $\mathbf{z}_{t+1} = \text{FastMix}\Big(\text{prox}_{\frac{\widehat{\gamma}}{1+\mu\widehat{\gamma}} g}\left(\frac{1}{1+\mu\widehat{\gamma}}\left(\mathbf{z}_t - \widehat{\gamma}\mathbf{s}_{t+1}\right)\right), K\Big)$;
  $\mathbf{y}_{t+1} = \text{FastMix}\left(\widehat{\tau}\mathbf{z}_{t+1} + (1-\widehat{\tau})\mathbf{y}_t, K\right)$.
end for

Similarly, the proximal mapping in Step 7 becomes $\text{prox}_{\widehat{\gamma}\widehat{g}}(\mathbf{z} - \widehat{\gamma}\mathbf{s}_{t+1})$. At the same time, it holds that

$$\begin{aligned}
&\text{prox}_{\widehat{\gamma}\widehat{g}}\left(\mathbf{z}_t^{(i)} - \widehat{\gamma}\mathbf{s}_{t+1}^{(i)}\right)\\
=&\operatorname*{argmin}_w\left(\widehat{g}(w) + \frac{1}{2\widehat{\gamma}}\left\|w - \left(\mathbf{z}_t^{(i)} - \widehat{\gamma}\mathbf{s}_{t+1}^{(i)}\right)\right\|^2\right)\\
=&\operatorname*{argmin}_w\left(g(w) + \frac{\mu}{2}\|w\|^2 + \frac{1}{2\widehat{\gamma}}\left\|w - \left(\mathbf{z}_t^{(i)} - \widehat{\gamma}\mathbf{s}_{t+1}^{(i)}\right)\right\|^2\right)\\
=&\operatorname*{argmin}_w\left(g(w) + \frac{1}{2\left(\frac{\widehat{\gamma}}{1+\mu\widehat{\gamma}}\right)}\left\|w - \frac{1}{\widehat{\gamma}}\frac{\widehat{\gamma}}{1+\mu\widehat{\gamma}}\left(\mathbf{z}_t^{(i)} - \widehat{\gamma}\mathbf{s}_{t+1}^{(i)}\right)\right\|^2\right)\\
=&\text{prox}_{\frac{\widehat{\gamma}}{1+\mu\widehat{\gamma}}g}\left(\frac{1}{1+\mu\widehat{\gamma}}\left(\mathbf{z}_t^{(i)} - \widehat{\gamma}\mathbf{s}_{t+1}^{(i)}\right)\right).
\end{aligned}$$

Thus, Step 7 of Algorithm 2 is replaced by $\mathbf{z}_{t+1} = \text{prox}_{\frac{\widehat{\gamma}}{1+\mu\widehat{\gamma}}g}\left(\frac{1}{1+\mu\widehat{\gamma}}\left(\mathbf{z} - \widehat{\gamma}\mathbf{s}_{t+1}\right)\right)$.

We give a detailed algorithm description in Algorithm 3 for solving the problems that $f_i(x)$ is L-smooth and μ-strongly convex while $g(x)$ is only convex. We provide the convergence rate of Algorithm 3 in the following theorem.

Theorem 5 *Let the objective function $F(x)$ be of the form (1) and each $f_i(x)$ be L-smooth and μ-strongly convex with $L \geq 2\mu$. Suppose that $g(x)$ is convex but may be non-smooth. Let us set the step size $\widehat{\gamma} = \frac{1}{20\sqrt{(L-\mu)\mu}}$, $\widehat{\tau} = \mu\widehat{\gamma}$, and $K = \frac{11}{\sqrt{1-\lambda_2(W)}}$ in Algorithm 3, then the output $\mathbf{z}_T$ satisfies that*

$$\begin{aligned}
\|\mathbf{z}_T - \mathbf{1}x^*\|^2 \leq &\left(1 - \frac{1}{40}\cdot\sqrt{\frac{\mu}{L-\mu}}\right)^T \cdot \left(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2\right.\\
&\left. + \frac{20^2(L-\mu)}{\mu}\cdot\left(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot(L-\mu)^2}\|\mathbf{\Pi}\mathbf{s}_0\|^2\right)\right).
\end{aligned} \tag{12}$$

Proof Let us denote that $\widehat{g}(x) = g(x) + \frac{\mu}{2}\|x\|^2$ and $\widehat{f}_i(x) = f_i(x) - \frac{\mu}{2}\|x\|^2$. Since $f_i(x)$ is L-smooth and μ-strongly convex, and $g(x)$ is only convex but may be not smooth, then $\widehat{g}(x)$

Algorithm 4 Multi-consensus

1: **Input:** $\mathbf{x}_0$, K.
2: **for** $k = 0, \ldots, K$ **do**
3: $\quad \mathbf{x}_{k+1} = W^{(k)}\mathbf{x}_k$;
4: **end for**
5: **Output:** $\mathbf{x}_K$.

is μ-strongly convex and $\widehat{f}_i(x)$ is $(L-\mu)$-smooth and convex which satisfy the conditions of Theorem 2. Running Algorithm 2 with $\widehat{g}(x)$ and $\widehat{f}(x)$ with a step size $\widehat{\gamma} = \frac{1}{20\sqrt{(L-\mu)\mu}}$ and $\widehat{\tau} = \mu\widehat{\gamma}$, then Theorem 2 guarantees the convergence rate in Eq. (12). ■

Remark 6 *By the similar analysis in the proof of Corollary 3, we can obtain Algorithm 3 can achieve a computation complexity is* $\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ *and a communication complexity* $\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\log\frac{1}{\epsilon}\right)$. *Thus, our work provides an optimal decentralized algorithm for solving problems that* $f_i(x)$ *is* L*-smooth and* μ*-strongly convex while* $g(x)$ *is only convex.*

3.4 Extension to Time-Varying Networks

We consider the decentralized composite optimization under the time-varying communication networks. When the topology of communication networks is time-varying, then the weight matrix W is not a constant matrix. Instead, it is a matrix depending on the current communication round k. Accordingly, we denote the weight matrix $W^{(k)}$. Next, we introduce the following assumption, which is the same as Kovalev et al. (2021).

Assumption 4 *A weight matrix* $W^{(k)} \in \mathbb{R}^{m\times m}$ *with* $W^{(k)}_{i,j} \neq 0$ *if and only if agents* i *and* j *are connected or* $i = j$. *It also satisfies that* $W^{(k)}\mathbf{1} = \mathbf{1}$, $\text{null}(I - W^{(k)}) = \text{span}(\mathbf{1})$. *There exists* $\chi \geq 1$ *for all* k, *such that*

$$\left\|W^{(k)}x - \frac{1}{m}\mathbf{1}\mathbf{1}^\top x\right\| \leq (1-\chi^{-1})\left\|x - \frac{1}{m}\mathbf{1}\mathbf{1}^\top x\right\|. \tag{13}$$

Note that, comparing Assumption 4 and Assumption 3, when the topology of the communication graph is time-varying, we do *not* require the weight matrix $W^{(k)}$ to be positive semi-definite, or even symmetric.

Next, we will extend Algorithm 2 to the setting that the topology of the communication graph is time-varying and the weight matrices $W^{(k)}$ satisfy Assumption 4. Because the varying weight matrices $W^{(k)}$ only affect the way to achieve the consensus between agents, we only replace the "FastMix" in Algorithm 1 by the "multi-consensus" shown in Algorithm 4. Accordingly, we propose our optimal decentralized accelerated proximal gradient descent for time-varying networks. The detailed algorithm description is listed in Algorithm 5.

Theorem 7 *Let the objective function* $F(x)$ *be of the form* (1) *and Assumption 1-2 hold. Assume that the topology of communication networks is time-varying and its weight matrices*

Algorithm 5 `ODAPG` for time-varying networks

Input: x_0, initial step size γ, τ.
Initialization: Set $\mathbf{z}_0 = \mathbf{y}_0 = \mathbf{x}_0 = \mathbf{1}x_0$, $\mathbf{s}_0^{(i)} = \nabla f_i(\mathbf{x}_0^{(i)})$, in parallel for $i \in [m]$.
for $t = 1, \ldots, T$ **do**
  Compute $\mathbf{x}_{t+1} = \tau\mathbf{z}_t + (1-\tau)\mathbf{y}_t$;
  Compute the local gradients $\nabla f_i(\mathbf{x}_{t+1}^{(i)})$ in parallel for $i \in [m]$ to form the gradient $\nabla\widetilde{f}(\mathbf{x}_{t+1})$;
  $\mathbf{s}_{t+1} = \text{Multi-consensus}(\mathbf{s}_t + \nabla\widetilde{f}(\mathbf{x}_{t+1}) - \nabla\widetilde{f}(\mathbf{x}_t), K)$;
  $\mathbf{z}_{t+1} = \text{Multi-consensus}\Big(\text{prox}_{\gamma g}\left(\mathbf{z}_t - \gamma\mathbf{s}_{t+1}\right), K\Big)$;
  $\mathbf{y}_{t+1} = \text{Multi-consensus}\left(\tau\mathbf{z}_{t+1} + (1-\tau)\mathbf{y}_t, K\right)$.
end for

$W^{(k)}$ satisfy Assumption 4. Letting us set the step size $\gamma = \frac{1}{20\sqrt{L\mu}}$, $\tau = \mu\gamma$, and $K = 11 \cdot \chi$ in Algorithm 5, letting x^* denote the minimum of $F(x)$, then it holds that

$$\begin{aligned}\|\mathbf{z}_T - \mathbf{1}x^*\|^2 \le \left(1 - \frac{1}{40}\cdot\sqrt{\frac{\mu}{L}}\right)^T \cdot \Bigg(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \|\mathbf{z}_0 - \mathbf{1}x^*\|^2 \\ + \frac{20^2 L}{2\mu}\cdot\left(\|\mathbf{\Pi}\mathbf{x}_0\|^2 + 8\|\mathbf{\Pi}\mathbf{y}_0\|^2 + \frac{9\mu}{50L}\|\mathbf{\Pi}\mathbf{z}_0\|^2 + \frac{3}{20^4\cdot L^2}\|\mathbf{\Pi}\mathbf{s}_0\|^2\right)\Bigg).\end{aligned} \tag{14}$$

Proof The only difference between Algorithm 2 and Algorithm 5 lies in the way to achieve the consensus. Comparing Eq. (13) with the equation in Proposition 1, we can obtain that it only requires $K = 11 \cdot \chi$ for "Multi-consensus" to achieve the same consensus for time-varying networks to which "FastMix" with $K = \frac{11}{\sqrt{1-\lambda_2(W)}}$ obtains for the static networks. ■

Next, by the convergence rate of our algorithm shown in Theorem 7, we will give our algorithm's computation cost and communication complexities in the following corollary.

Corollary 8 *Let the objective function and communication networks satisfy the property described in Theorem 7. To find an ϵ-suboptimal solution, the computation and communication complexities of Algorithm 5 for each agent are*

$$T = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right), \quad \textit{and} \quad C = \mathcal{O}\left(\chi\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right). \tag{15}$$

Proof The proof of computation complexity is the same as that of Corollary 3. For the communication complexity, we have

$$C = 3TK = T\cdot 33\cdot\chi = \mathcal{O}\left(\chi\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right).$$

■

Remark 9 *Eq.* (15) *demonstrates that Algorithm 5 can match the lower bounds of smooth and strongly convex decentralized optimization over time-varying networks (Kovalev et al., 2021). Hence, our algorithm realizes the optimal benchmarks in both computational and communication complexities.*

4. Convergence Analysis

In this section, we will prove Theorem 2. First, we will provide important lemmas related to "accelerated proximal gradient descent". Next, we will bound the consensus error terms. Finally, we will provide the detailed proof of Theorem 2.

4.1 Analysis Related to Nesterov's Acceleration

First, we give the relation between average variables in the following lemma.

Lemma 10 *Letting $\overline{x}_t$, $\overline{y}_t$ and $\overline{z}_t$ (refer to Eq.* (2)*) be the average variables of $\mathbf{x}_t$, $\mathbf{y}_t$, $\mathbf{z}_t$ respectively, then it holds that*

$$\overline{x}_{t+1} = \tau\overline{z}_t + (1-\tau)\overline{y}_t, \tag{16}$$

$$\overline{y}_{t+1} = \tau\overline{z}_{t+1} + (1-\tau)\overline{y}_t, \tag{17}$$

$$\overline{s}_t = \frac{1}{m}\sum_{i=1}^{m}\nabla f_i(\mathbf{x}_t^{(i)}). \tag{18}$$

Proof By multiplying $\frac{1}{m}\mathbf{1}^\top$ both sides of Eq. (6), (7) and (9) respectively and using the definition of average variables defined in Eq. (2), we can obtain the result. ■

Next, under Assumption 1, we give two properties similar to the ones of convexity and smoothness with the inexact gradient $\overline{s}_t$.

Lemma 11 (Lemma 1 of Li and Lin (2021)) *Define*

$$f(\overline{x}_t, \mathbf{x}_t) \triangleq \frac{1}{m}\sum_{i=1}^{m}\left(f_i(\mathbf{x}_t^{(i)}) + \left\langle \nabla f_i(\mathbf{x}_t^{(i)}), \overline{x}_t - \mathbf{x}_t^{(i)}\right\rangle\right) \tag{19}$$

Supposing Assumption 1 holds, then we have for any $w \in \mathbb{R}^d$,

$$f(w) \geq f(\overline{x}_t, \mathbf{x}_t) + \langle \overline{s}_t, w - \overline{x}_t\rangle, \tag{20}$$

$$f(w) \leq f(\overline{x}_t, \mathbf{x}_t) + \langle \overline{s}_t, w - \overline{x}_t\rangle + \frac{L}{2}\|w - \overline{x}_t\|^2 + \frac{L}{2m}\|\mathbf{\Pi}\mathbf{x}_t\|^2. \tag{21}$$

Next, we construct a lower bound on the one-iteration progress of Algorithm 2.

Lemma 12 *Letting Assumption 1 hold, then sequences $\{\mathbf{x}_t\}$, $\{\mathbf{y}_t\}$, $\{\mathbf{z}_t\}$ and $\{\mathbf{s}_t\}$ generated by Algorithm 2 satisfy the following property*

$$\begin{aligned}\gamma\left(f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) - f(x^*)\right) \leq & \frac{\gamma(1-\tau)}{\tau}\Big(f(\overline{y}_t) - f(\overline{x}_{t+1}, \mathbf{x}_{t+1})\Big) - \frac{\gamma(1-\tau)}{\tau}D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\ & + \frac{\gamma}{\tau}\left\langle \overline{s}_{t+1}, \overline{x}_{t+1} - \overline{y}_{t+1}\right\rangle + \gamma\left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^*\right\rangle,\end{aligned} \tag{22}$$

where $f(\overline{x}_{t+1}, \mathbf{x}_{t+1})$ and $D_f(\overline{y}_t, \mathbf{x}_{t+1})$ are defined in Eq. (19) *and Eq.* (5)*, respectively.*

Proof By the convexity of $f_i(\cdot)$ and update rule of Algorithm 2, we can obtain

$$
\begin{aligned}
&\gamma \left(f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) - f(x^*)\right) \\
&\overset{(20)}{\leq} \gamma \left\langle \overline{s}_{t+1}, \overline{x}_{t+1} - x^* \right\rangle \\
&= \gamma \left\langle \overline{s}_{t+1}, \overline{x}_{t+1} - \overline{z}_t \right\rangle + \gamma \left\langle \overline{s}_{t+1}, \overline{z}_t - x^* \right\rangle \\
&\overset{(16)}{=} \frac{\gamma(1-\tau)}{\tau} \left\langle \overline{s}_{t+1}, \overline{y}_t - \overline{x}_{t+1} \right\rangle + \gamma \left\langle \overline{s}_{t+1}, \overline{z}_t - x^* \right\rangle \\
&\overset{(19)(5)(18)}{=} \frac{\gamma(1-\tau)}{\tau} \Big(f(\overline{y}_t) - f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) - D_f(\overline{y}_t, \mathbf{x}_{t+1}) \Big) + \gamma \left\langle \overline{s}_{t+1}, \overline{z}_t - x^* \right\rangle \\
&= \frac{\gamma(1-\tau)}{\tau} \Big(f(\overline{y}_t) - f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) \Big) - \frac{\gamma(1-\tau)}{\tau} D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\
&+ \gamma \left\langle \overline{s}_{t+1}, \overline{z}_t - \overline{z}_{t+1} \right\rangle + \gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle \\
&\overset{(16)(17)}{=} \frac{\gamma(1-\tau)}{\tau} \Big(f(\overline{y}_t) - f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) \Big) - \frac{\gamma(1-\tau)}{\tau} D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\
&+ \frac{\gamma}{\tau} \left\langle \overline{s}_{t+1}, \overline{x}_{t+1} - \overline{y}_{t+1} \right\rangle + \gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle .
\end{aligned}
$$

■

Moreover, we bound the value of $\frac{\gamma}{\tau} \left\langle \overline{s}_{t+1}, \overline{x}_{t+1} - \overline{y}_{t+1} \right\rangle$ and $\gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle$ in the next two lemmas.

Lemma 13 *Letting Assumption 2 hold, that is, $g(\cdot)$ is μ-strongly convex, then it holds that*

$$
\begin{aligned}
&\gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle \\
\leq &\frac{1}{2m} \left\| \mathbf{z}_t - \mathbf{1}x^* \right\|^2 - \frac{1+\mu\gamma}{2m} \left\| \mathbf{z}_{t+1} - \mathbf{1}x^* \right\|^2 - \frac{1}{2\tau^2} \left\| \overline{x}_{t+1} - \overline{y}_{t+1} \right\|^2 + \gamma g(x^*) \\
&- \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} g(\mathbf{z}_{t+1}^{(i)}) + \frac{1}{2m} \left(\gamma^2 \left\| \mathbf{\Pi} \mathbf{s}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 \right).
\end{aligned} \tag{23}
$$

Proof By Lemma 25 with $z = \mathbf{z}_{t+1}^{(i)}$, $x = \mathbf{z}_t^{(i)}$, $d = \mathbf{s}_{t+1}^{(i)}$, $y = x^*$ and $\eta = \gamma$, we have

$$
\begin{aligned}
\gamma \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle \leq &\frac{1}{2} \left\| \mathbf{z}_t^{(i)} - x^* \right\|^2 - \frac{1+\mu\gamma}{2} \left\| \mathbf{z}_{t+1}^{(i)} - x^* \right\| - \frac{1}{2} \left\| \mathbf{z}_{t+1}^{(i)} - \mathbf{z}_t^{(i)} \right\|^2 \\
&- \gamma g(\mathbf{z}_{t+1}^{(i)}) + \gamma g(x^*).
\end{aligned}
$$

We also have

$$
\begin{aligned}
&\gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle \\
=&\gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle + \gamma \left(\left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle - \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle \right) \\
=&\gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle + \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1} \right\rangle \\
=&\gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle + \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)} - \overline{s}_{t+1}, \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1} \right\rangle \\
\overset{(31)}{\leq}& \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle + \frac{1}{m} \sum_{i=1}^{m} \left(\frac{\gamma^2 \left\| \mathbf{s}_{t+1}^{(i)} - \overline{s}_{t+1} \right\|^2 + \left\| \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1} \right\|^2}{2} \right) \\
=&\gamma \cdot \frac{1}{m} \sum_{i=1}^{m} \left\langle \mathbf{s}_{t+1}^{(i)}, \mathbf{z}_{t+1}^{(i)} - x^* \right\rangle + \frac{1}{2m} \left(\gamma^2 \left\| \mathbf{\Pi} \mathbf{s}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 \right).
\end{aligned}
$$

Using the above two equations, we can obtain that

$$
\begin{aligned}
\gamma \left\langle \overline{s}_{t+1}, \overline{z}_{t+1} - x^* \right\rangle \leq& \frac{1}{2m} \left\| \mathbf{z}_t - \mathbf{1} x^* \right\|^2 - \frac{1 + \mu\gamma}{2m} \left\| \mathbf{z}_{t+1} - \mathbf{1} x^* \right\|^2 - \frac{1}{2m} \left\| \mathbf{z}_{t+1} - \mathbf{z}_t \right\|^2 \\
&- \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} g(\mathbf{z}_{t+1}^{(i)}) + \gamma g(x^*) + \frac{1}{2m} \left(\gamma^2 \left\| \mathbf{\Pi} \mathbf{s}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 \right).
\end{aligned}
$$

Furthermore,

$$
\begin{aligned}
&- \left\| \mathbf{z}_{t+1} - \mathbf{z}_t \right\|^2 \\
=& - \left\| \mathbf{z}_{t+1} - \mathbf{1} \overline{z}_{t+1} - (\mathbf{z}_t - \mathbf{1} \overline{z}_t) + (\mathbf{1} \overline{z}_{t+1} - \mathbf{1} \overline{z}_t) \right\|^2 \\
=& - \left(\left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_t \right\|^2 + m \left\| \overline{z}_{t+1} - \overline{z}_t \right\|^2 \right) \\
&- 2 \sum_{i=1}^{m} \left\langle \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1} - (\mathbf{z}_t^{(i)} - \overline{z}_t), \overline{z}_{t+1} - \overline{z}_t \right\rangle \\
&+ 2 \sum_{i=1}^{m} \left\langle \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1}, \mathbf{z}_t^{(i)} - \overline{z}_t \right\rangle \\
=& - \left(\left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_t \right\|^2 + m \left\| \overline{z}_{t+1} - \overline{z}_t \right\|^2 \right) + 2 \sum_{i=1}^{m} \left\langle \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1}, \mathbf{z}_t^{(i)} - \overline{z}_t \right\rangle \\
\overset{(31)}{\leq}& - \left(\left\| \mathbf{\Pi} \mathbf{z}_{t+1} \right\|^2 + \left\| \mathbf{\Pi} \mathbf{z}_t \right\|^2 + m \left\| \overline{z}_{t+1} - \overline{z}_t \right\|^2 \right) + \sum_{i=1}^{m} \left(\left\| \mathbf{z}_{t+1}^{(i)} - \overline{z}_{t+1} \right\|^2 + \left\| \mathbf{z}_t^{(i)} - \overline{z}_t \right\|^2 \right) \\
=& m \left\| \overline{z}_{t+1} - \overline{z}_t \right\|^2 \overset{(16)(17)}{=} \frac{m}{\tau^2} \left\| \overline{x}_{t+1} - \overline{y}_{t+1} \right\|^2,
\end{aligned}
$$

where the third equality is because of

$$\begin{aligned}&\sum_{i=1}^{m}\left\langle \mathbf{z}_{t+1}^{(i)}-\overline{z}_{t+1}-(\mathbf{z}_{t}^{(i)}-\overline{z}_{t}),\overline{z}_{t+1}-\overline{z}_{t}\right\rangle\\=&\left\langle \sum_{i=1}^{m}\mathbf{z}_{t+1}^{(i)}-m\overline{z}_{t+1}+\sum_{i=1}^{m}\mathbf{z}_{t}^{(i)}-m\overline{z}_{t},\overline{z}_{t+1}-\overline{z}_{t}\right\rangle=0.\end{aligned}$$

Therefore, we can obtain that

$$\begin{aligned}&\gamma\left\langle \overline{s}_{t+1},\overline{z}_{t+1}-x^{*}\right\rangle\\\leq&\frac{1}{2m}\left\|\mathbf{z}_{t}-\mathbf{1}x^{*}\right\|^{2}-\frac{1+\mu\gamma}{2m}\left\|\mathbf{z}_{t+1}-\mathbf{1}x^{*}\right\|^{2}-\frac{1}{2\tau^{2}}\left\|\overline{x}_{t+1}-\overline{y}_{t+1}\right\|^{2}+\gamma g(x^{*})\\&-\gamma\cdot\frac{1}{m}\sum_{i=1}^{m}g(\mathbf{z}_{t+1}^{(i)})+\frac{1}{2m}\left(\gamma^{2}\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^{2}+\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^{2}\right).\end{aligned}$$

■

Lemma 14 *Letting Assumption 1 and Assumption 2 hold, then it holds that*

$$\begin{aligned}\frac{\gamma}{\tau}\left\langle \overline{s}_{t+1},\overline{x}_{t+1}-\overline{y}_{t+1}\right\rangle\leq&\frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1},\mathbf{x}_{t+1})-F(\overline{y}_{t+1})\Big)+\frac{\gamma}{m}\sum_{i=1}^{m}g(\mathbf{z}_{t+1}^{(i)})+\frac{\gamma(1-\tau)}{\tau}g(\overline{y}_{t})\\&+\frac{L\gamma}{2\tau}\left\|\overline{y}_{t+1}-\overline{x}_{t+1}\right\|^{2}+\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^{2}.\end{aligned}\tag{24}$$

Proof By the L-smoothness of $f_i(\cdot)$ and the convexity of $g(\cdot)$, we have

$$\begin{aligned}&\frac{\gamma}{\tau}\left\langle \overline{s}_{t+1},\overline{x}_{t+1}-\overline{y}_{t+1}\right\rangle\\\overset{(21)}{\leq}&\frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1},\mathbf{x}_{t+1})-f(\overline{y}_{t+1})\Big)+\frac{L\gamma}{2\tau}\left\|\overline{y}_{t+1}-\overline{x}_{t+1}\right\|^{2}+\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^{2}\\=&\frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1},\mathbf{x}_{t+1})-F(\overline{y}_{t+1})\Big)+\frac{\gamma}{\tau}g(\overline{y}_{t+1})+\frac{L\gamma}{2\tau}\left\|\overline{y}_{t+1}-\overline{x}_{t+1}\right\|^{2}+\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^{2}\\\leq&\frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1},\mathbf{x}_{t+1})-F(\overline{y}_{t+1})\Big)+\gamma g(\overline{z}_{t+1})+\frac{\gamma(1-\tau)}{\tau}g(\overline{y}_{t})\\&+\frac{L\gamma}{2\tau}\left\|\overline{y}_{t+1}-\overline{x}_{t+1}\right\|^{2}+\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^{2}\\\leq&\frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1},\mathbf{x}_{t+1})-F(\overline{y}_{t+1})\Big)+\frac{\gamma}{m}\sum_{i=1}^{m}g(\mathbf{z}_{t+1}^{(i)})+\frac{\gamma(1-\tau)}{\tau}g(\overline{y}_{t})\\&+\frac{L\gamma}{2\tau}\left\|\overline{y}_{t+1}-\overline{x}_{t+1}\right\|^{2}+\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^{2},\end{aligned}$$

where the second inequality is because of Eq. (17) and the convexity of $g(\cdot)$. The last inequality is because of $\overline{z}_{t+1}=\frac{1}{m}\sum_{i=1}^{m}\mathbf{z}_{t+1}^{(i)}$ and the convexity of $g(\cdot)$. ■

Lemma 15 *Letting Assumption 1 and Assumption 2 hold, then it holds that*

$$\begin{aligned}0 \leq & \frac{\gamma(1-\tau)}{\tau} F(\overline{y}_t) - \frac{\gamma}{\tau} F(\overline{y}_{t+1}) + \gamma F(x^*) + \left(\frac{L\gamma}{2\tau} - \frac{1}{2\tau^2}\right) \left\|\overline{x}_{t+1} - \overline{y}_{t+1}\right\|^2 \\ & + \frac{1}{2m} \left\|\mathbf{z}_t - \mathbf{1}x^*\right\|^2 - \frac{1+\mu\gamma}{2m} \left\|\mathbf{z}_{t+1} - \mathbf{1}x^*\right\|^2 - \frac{\gamma(1-\tau)}{\tau} D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\ & + \frac{L\gamma}{2m\tau} \left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 + \frac{1}{2m}\left(\gamma^2 \left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + \left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2\right).\end{aligned} \tag{25}$$

Proof Combining Eq. (22), Eq. (23) and Eq. (24), we can obtain that

$$\begin{aligned}0 \leq & -\gamma\Big(f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) - f(x^*)\Big) + \frac{\gamma(1-\tau)}{\tau}\Big(f(\overline{y}_t) - f(\overline{x}_{t+1}, \mathbf{x}_{t+1})\Big) - \frac{\gamma(1-\tau)}{\tau} D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\ & + \frac{1}{2m} \left\|\mathbf{z}_t - \mathbf{1}x^*\right\|^2 - \frac{1+\mu\gamma}{2} \left\|\mathbf{z}_{t+1} - \mathbf{1}x^*\right\|^2 - \frac{1}{2\tau^2} \left\|\overline{x}_{t+1} - \overline{y}_{t+1}\right\|^2 + \gamma g(x^*) \\ & - \gamma \cdot \frac{1}{m} \sum_{i=1}^{m} g(\mathbf{z}_{t+1}^{(i)}) + \frac{1}{2m}\left(\gamma^2 \left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + \left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2\right) \\ & + \frac{\gamma}{\tau}\Big(f(\overline{x}_{t+1}, \mathbf{x}_{t+1}) - F(\overline{y}_{t+1})\Big) + \frac{\gamma}{m} \sum_{i=1}^{m} g(\mathbf{z}_{t+1}^{(i)}) + \frac{\gamma(1-\tau)}{\tau} g(\overline{y}_t) \\ & + \frac{L\gamma}{2\tau} \left\|\overline{y}_{t+1} - \overline{x}_{t+1}\right\|^2 + \frac{L\gamma}{2m\tau} \left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 \\ = & \frac{\gamma(1-\tau)}{\tau} F(\overline{y}_t) - \frac{\gamma}{\tau} F(\overline{y}_{t+1}) + \gamma F(x^*) + \left(\frac{L\gamma}{2\tau} - \frac{1}{2\tau^2}\right) \left\|\overline{x}_{t+1} - \overline{y}_{t+1}\right\|^2 \\ & + \frac{1}{2m} \left\|\mathbf{z}_t - \mathbf{1}x^*\right\|^2 - \frac{1+\mu\gamma}{2} \left\|\mathbf{z}_{t+1} - \mathbf{1}x^*\right\|^2 - \frac{\gamma(1-\tau)}{\tau} D_f(\overline{y}_t, \mathbf{x}_{t+1}) \\ & + \frac{L\gamma}{2m\tau} \left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 + \frac{1}{2m}\left(\gamma^2 \left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + \left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2\right).\end{aligned}$$

■

Remark 16 *Lemma 15 shares almost the same result as the one of Lemma 4 of Driggs et al. (2022). This is because the update form of gradient tracking is almost the same as the one of SARAH (Nguyen et al., 2017), which is a variance reduction method. Accordingly, our main proof follows from proof framework of Driggs et al. (2022) which is used to prove the convergence rate of accelerated variance reduction methods with proximal mapping. However, our proof has several differences from the one of Driggs et al. (2022). First, our proof uses the convexity and smoothness properties with respect to the inexact gradient $\overline{s}_t$ (refer to Eq.*(20) *and* (21)*) instead of the standard ones. Moreover, in our lemma, we have extra consensus terms $\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2$, $\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2$ and $\gamma^2 \left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2$ which are because of the decentralized optimization setting. Furthermore, an extended Bregman divergence $D_f(\overline{y}_t, \mathbf{x}_{t+1})$ is used in our lemma while the standard Bregman divergence is used in Lemma 4 of Driggs et al. (2022).*

Remark 17 *Though our algorithm is almost the same to the one of Li and Lin (2021) except the difference in the proximal mapping in the Eq.* (8)*, its proof framework can not be*

extended to solve the composite optimization problems. This is because Li and Lin (2021) can not bound the error terms caused by the proximal mapping in their proof framework.

4.2 Consensus Error Bound

In this section, we will bound the consensus error terms in Lemma 15. First, we prove a property related to the proximal mapping in the decentralized setting.

Lemma 18 *For any $\mathbf{x} \in \mathbb{R}^{m\times d}$, the proximal mapping $\mathrm{prox}_{\gamma g}$ has the following property,*

$$\left\|\mathrm{prox}_{\gamma g}\left(\frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\right) - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathrm{prox}_{\gamma g}(\mathbf{x})\right\| \le \|\mathbf{\Pi}\mathbf{x}\|. \tag{26}$$

Proof Using the non-expansiveness of the proximal mapping, we have

$$\begin{aligned}
&\left\|\mathrm{prox}_{\gamma g}\left(\frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\right) - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathrm{prox}_{\gamma g}(\mathbf{x})\right\|^2 = m\left\|\mathrm{prox}_{\gamma g}\left(\frac{1}{m}\mathbf{1}^\top\mathbf{x}\right) - \frac{1}{m}\sum_{i=1}^m \mathrm{prox}_{\gamma g}(\mathbf{x}^{(i)})\right\|^2 \\
=&m\left\|\frac{1}{m}\sum_{i=1}^m\left(\mathrm{prox}_{\gamma g}\left(\frac{1}{m}\mathbf{1}^\top\mathbf{x}\right) - \mathrm{prox}_{\gamma g}(\mathbf{x}^{(i)})\right)\right\|^2 \le \sum_{i=1}^m\left\|\mathrm{prox}_{\gamma g}\left(\frac{1}{m}\mathbf{1}^\top\mathbf{x}\right) - \mathrm{prox}_{\gamma g}(\mathbf{x}^{(i)})\right\|^2 \\
\le&\sum_{i=1}^m\left\|\frac{1}{m}\mathbf{1}^\top\mathbf{x} - \mathbf{x}^{(i)}\right\|^2 = \left\|\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\right\|^2 = \|\mathbf{\Pi}\mathbf{x}\|^2.
\end{aligned}$$

■

Next, we will give the upper bounds of consensus error terms and their convergence properties.

Lemma 19 *Letting $K = \frac{11}{\sqrt{1-\lambda_2(W)}}$, and the step size γ be properly chosen such that $L^2\gamma^2 \le \frac{1}{32\cdot 73\cdot\tau^2}$ in Algorithm 2, then it holds that*

$$\begin{aligned}
\|\mathbf{\Pi}\mathbf{x}_{t+1}\|^2 \le& 2\|\mathbf{\Pi}\mathbf{y}_t\|^2 + 2\tau^2\|\mathbf{\Pi}\mathbf{z}_t\|^2, \qquad \|\mathbf{\Pi}\mathbf{z}_{t+1}\|^2 \le \frac{1}{4}\left(\|\mathbf{\Pi}\mathbf{z}_t\|^2 + \gamma^2\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2\right), \\
\gamma^2\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2 \le& \frac{1}{16}\cdot\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2 + 8\gamma^2\left(mLD_f(\overline{y}_t, \mathbf{x}_{t+1}) + mL^2\|\overline{x}_t - \overline{y}_t\|^2 + L^2\|\mathbf{\Pi}\mathbf{x}_t\|^2\right),
\end{aligned} \tag{27}$$

and

$$\begin{aligned}
&\|\mathbf{\Pi}\mathbf{x}_{t+1}\|^2 + c_1\|\mathbf{\Pi}\mathbf{y}_{t+1}\|^2 + c_2\|\mathbf{\Pi}\mathbf{z}_{t+1}\|^2 + c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2 \\
&\le \frac{1}{2}\left(\|\mathbf{\Pi}\mathbf{x}_t\|^2 + c_1\|\mathbf{\Pi}\mathbf{y}_t\|^2 + c_2\|\mathbf{\Pi}\mathbf{z}_t\|^2 + c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2\right) \\
&+ c_4\left(mL\gamma^2 D_f(\overline{y}_t, \mathbf{x}_{t+1}) + mL^2\gamma^2\|\overline{x}_t - \overline{y}_t\|^2\right),
\end{aligned} \tag{28}$$

with $c_1 = 8$, $c_2 = 72\tau^2$, $c_3 = 3\tau^2$, $c_4 = 168\tau^2$.

Proof For the notation convenience, we denote that

$$\mathbb{T}(\mathbf{x}) \triangleq \mathrm{FastMix}(\mathbf{x}, K).$$

By the chosen communication step number K in "FastMix" and Proposition 1, we can obtain that

$$\|\mathbf{\Pi}\cdot\mathbb{T}(\mathbf{x})\|\leq\rho\,\|\mathbf{\Pi}\mathbf{x}\|\,,\ \text{with}\ \rho^2\leq\frac{1}{32}.$$

First, by the update rule of $\mathbf{z}_{t+1}$ in Eq (8), we can obtain

$$\|\mathbf{\Pi}\mathbf{z}_{t+1}\|=\left\|\mathbf{\Pi}\cdot\mathbb{T}\big(\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\big)\right\|\leq\rho\left\|\mathbf{\Pi}\cdot\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\right\|.$$

Furthermore,

$$\begin{aligned}
&\left\|\mathbf{\Pi}\cdot\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\right\|=\left\|\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})-\frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\right\|\\
&\leq\left\|\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})-\mathrm{prox}_{\gamma g}(\mathbf{1}(\overline{z}_t-\gamma\overline{s}_{t+1}))\right\|\\
&+\left\|\mathrm{prox}_{\gamma g}(\mathbf{1}(\overline{z}_t-\gamma\overline{s}_{t+1}))-\frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\right\|\\
&\leq\|\mathbf{z}_t-\mathbf{1}\overline{z}_t\|+\gamma\,\|\mathbf{s}_{t+1}-\mathbf{1}\overline{s}_{t+1}\|+\left\|\mathrm{prox}_{\gamma g}(\mathbf{1}(\overline{z}_t-\gamma\overline{s}_{t+1}))-\frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathrm{prox}_{\gamma g}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\right\|\\
&\overset{(26)}{\leq}\|\mathbf{\Pi}\mathbf{z}_t\|+\gamma\,\|\mathbf{\Pi}\mathbf{s}_{t+1}\|+\|\mathbf{\Pi}(\mathbf{z}_t-\gamma\mathbf{s}_{t+1})\|\\
&\leq 2\,\|\mathbf{\Pi}\mathbf{z}_t\|+2\gamma\,\|\mathbf{\Pi}\mathbf{s}_{t+1}\|\,,
\end{aligned}$$

where the second inequality is because of the non-expansiveness of the proximal mapping. Thus, it holds that

$$\|\mathbf{\Pi}\mathbf{z}_{t+1}\|^2\leq\rho^2\big(2\,\|\mathbf{\Pi}\mathbf{z}_t\|+2\gamma\,\|\mathbf{\Pi}\mathbf{s}_{t+1}\|\,\big)^2\leq 8\rho^2\left(\|\mathbf{\Pi}\mathbf{z}_t\|^2+\gamma^2\,\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2\right).$$

Furthermore, by the update rule of $\mathbf{x}_{t+1}$ and $\mathbf{y}_{t+1}$ in Eq. (6) and (9) respectively, we have

$$\begin{aligned}
\|\mathbf{\Pi}\mathbf{x}_{t+1}\|^2&\leq(\tau\,\|\mathbf{\Pi}\mathbf{z}_t\|+(1-\tau)\,\|\mathbf{\Pi}\mathbf{y}_t\|)^2\leq 2\tau^2\,\|\mathbf{\Pi}\mathbf{z}_t\|^2+2\,\|\mathbf{\Pi}\mathbf{y}_t\|^2\\
\|\mathbf{\Pi}\mathbf{y}_{t+1}\|^2&\leq(\tau\,\|\mathbf{\Pi}\mathbf{z}_t\|+\rho(1-\tau)\,\|\mathbf{\Pi}\mathbf{y}_t\|)^2\leq 2\tau^2\,\|\mathbf{\Pi}\mathbf{z}_t\|^2+2\rho^2\,\|\mathbf{\Pi}\mathbf{y}_t\|^2.
\end{aligned}$$

By the update rule of $\mathbf{s}_{t+1}$ in Eq. (7), we have

$$\begin{aligned}
\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2=&\left\|\mathbf{\Pi}\cdot\mathbb{T}\big(\mathbf{s}_t+\nabla\widetilde{f}(\mathbf{x}_{t+1})-\nabla\widetilde{f}(\mathbf{x}_t)\big)\right\|^2\\
\leq&\rho^2\left\|\mathbf{\Pi}\left(\mathbf{s}_t+\nabla\widetilde{f}(\mathbf{x}_{t+1})-\nabla\widetilde{f}(\mathbf{x}_t)\right)\right\|^2\\
\leq&2\rho^2\,\|\mathbf{\Pi}\mathbf{s}_t\|^2+2\left\|\nabla\widetilde{f}(\mathbf{x}_{t+1})-\nabla\widetilde{f}(\mathbf{x}_t)\right\|^2.
\end{aligned}$$

Furthermore,

$$\begin{aligned}
&\left\|\nabla\widetilde{f}(\mathbf{x}_{t+1})-\nabla\widetilde{f}(\mathbf{x}_t)\right\|^2=\sum_{i=1}^m\left\|\nabla f_i(\mathbf{x}_{t+1}^{(i)})-\nabla f_i(\mathbf{x}_t^{(i)})\right\|^2\\
&\overset{(32)}{\leq} 2\sum_{i=1}^m\left\|\nabla f_i(\mathbf{x}_{t+1}^{(i)})-\nabla f_i(\overline{y}_t)\right\|^2+4\sum_{i=1}^m\left(\|\nabla f_i(\overline{y}_t)-\nabla f_i(\overline{x}_t)\|^2+\left\|\nabla f_i(\overline{x}_t)-\nabla f_i(\mathbf{x}_t^{(i)})\right\|^2\right)\\
&\overset{(36)(5)}{\leq} 4mLD_f(\overline{y}_t,\mathbf{x}_{t+1})+4\sum_{i=1}^m\left(\|\nabla f_i(\overline{y}_t)-\nabla f_i(\overline{x}_t)\|^2+\left\|\nabla f_i(\overline{x}_t)-\nabla f_i(\mathbf{x}_t^{(i)})\right\|^2\right)\\
&\overset{(37)}{\leq} 4mLD_f(\overline{y}_t,\mathbf{x}_{t+1})+4L^2\sum_{i=1}^m\left(\|\overline{x}_t-\overline{y}_t\|^2+\left\|\mathbf{x}_t^{(i)}-\overline{x}_t\right\|^2\right)\\
&=4mLD_f(\overline{y}_t,\mathbf{x}_{t+1})+4mL^2\|\overline{x}_t-\overline{y}_t\|^2+4L^2\|\mathbf{\Pi}\mathbf{x}_t\|^2.
\end{aligned}$$

Thus, it holds that

$$\gamma^2\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2\leq 2\rho^2\cdot\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2+8\gamma^2\left(mLD_f(\overline{y}_t,\mathbf{x}_{t+1})+mL^2\|\overline{x}_t-\overline{y}_t\|^2+L^2\|\mathbf{\Pi}\mathbf{x}_t\|^2\right).$$

Combining the above results, we can obtain that

$$\begin{aligned}
&\|\mathbf{\Pi}\mathbf{x}_{t+1}\|^2+c_1\|\mathbf{\Pi}\mathbf{y}_{t+1}\|^2+c_2\|\mathbf{\Pi}\mathbf{z}_{t+1}\|^2+c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_{t+1}\|^2\\
&\leq\left(64c_2\rho^2+8c_3\right)L^2\gamma^2\cdot\|\mathbf{\Pi}\mathbf{x}_t\|^2+\left(\frac{2}{c_1}+2\rho^2\right)\cdot c_1\|\mathbf{\Pi}\mathbf{y}_t\|^2+\left(\frac{2\tau^2}{c_2}+\frac{2c_1\tau^2}{c_2}+8\rho^2\right)\cdot c_2\|\mathbf{\Pi}\mathbf{z}_t\|^2\\
&+\left(\frac{16\rho^2c_2}{c_3}+2\right)\rho^2\cdot c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2+8\left(8\rho^2c_2+c_3\right)\left(mL\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+mL^2\gamma^2\|\overline{x}_t-\overline{y}_t\|^2\right)\\
&\leq(2c_2+8c_3)L^2\gamma^2\cdot\|\mathbf{\Pi}\mathbf{x}_t\|^2+\left(\frac{1}{4}+\frac{1}{16}\right)\cdot c_1\|\mathbf{\Pi}\mathbf{y}_t\|^2+\left(\frac{18\tau^2}{72\tau^2}+\frac{8}{32}\right)\cdot c_2\|\mathbf{\Pi}\mathbf{z}_t\|^2\\
&+\left(\frac{72\tau^2}{2\cdot3\tau^2}+2\right)\frac{1}{32}\cdot c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2\\
&+8\left(\frac{8\cdot72\tau^2}{32}c_2+3\tau^2\right)\left(mL\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+mL^2\gamma^2\|\overline{x}_t-\overline{y}_t\|^2\right)\\
&\leq\frac{1}{2}\left(\|\mathbf{\Pi}\mathbf{x}_t\|^2+c_1\|\mathbf{\Pi}\mathbf{y}_t\|^2+c_2\|\mathbf{\Pi}\mathbf{z}_t\|^2+c_3\gamma^2\|\mathbf{\Pi}\mathbf{s}_t\|^2\right)\\
&+c_4\left(mL\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+mL^2\gamma^2\|\overline{x}_t-\overline{y}_t\|^2\right),
\end{aligned}$$

where the last inequalities are because of $\rho^2\leq\frac{1}{32}$, the value of c_1, c_2, c_3, and c_4, and the condition $L^2\gamma^2\leq\frac{1}{32\cdot73\cdot\tau^2}=\frac{1}{2(2c_2+8c_3)}$. ■

Remark 20 *The proof of Lemma 19 partly follows the one of Li and Lin (2021). The error terms in Lemma 19 depends on $D_f(\overline{y}_t,\mathbf{x}_{t+1})$ which is the same to those of Li and Lin (2021). However, due to the proximal terms in the composite optimization, our proof takes extra steps to tackle the errors caused by proximal mappings.*

Next, we will bound the consensus error terms in Eq. (25) and sum them from $t = 0$ to $T - 1$ scaled with a non-negative sequence $\{\beta_t\}$.

Lemma 21 *Let $K = \frac{11}{\sqrt{1-\lambda_2(W)}}$, and the step size γ be properly chosen such that $L^2\gamma^2 \leq \frac{1}{32\cdot 73\cdot \tau^2}$ in Algorithm 2. Given a non-negative sequence $\{\beta_t\}$ with $t = 0, \ldots, T-1$, then it holds that*

$$
\begin{aligned}
&\sum_{t=0}^{T-1} \beta_t \left(\frac{L\gamma}{2m\tau} \left\| \mathbf{\Pi x}_{t+1} \right\|^2 + \frac{1}{2m} \left(\gamma^2 \left\| \mathbf{\Pi s}_{t+1} \right\|^2 + \left\| \mathbf{\Pi z}_{t+1} \right\|^2 \right) \right) \\
&\leq \frac{4c_5}{m} \left(\left\| \mathbf{\Pi x}_0 \right\|^2 + c_1 \left\| \mathbf{\Pi y}_0 \right\|^2 + c_2 \left\| \mathbf{\Pi z}_0 \right\|^2 + c_3\gamma^2 \left\| \mathbf{\Pi s}_0 \right\|^2 \right) \cdot \sum_{t=0}^{T-1} \beta_t 2^{-t} \\
&+ 8(1 + c_4c_5) \sum_{t=0}^{T-2} \beta_t \left(L\gamma^2 D_f(\overline{y}_t, \mathbf{x}_{t+1}) + L^2\gamma^2 \left\| \overline{x}_t - \overline{y}_t \right\|^2 \right) \\
&+ \beta_{T-1} \left(8L\gamma^2 D_f(\overline{y}_{T-1}, \mathbf{x}_T) + 8L^2\gamma^2 \left\| \overline{x}_{T-1} - \overline{y}_{T-1} \right\|^2 \right).
\end{aligned}
$$

where $c_5 \triangleq \max\left\{ 8L^2\gamma^2, \frac{L\gamma}{\tau c_1}, \frac{2}{c_2}, \frac{1}{16c_3} \right\}$ and c_1, c_2, c_3, c_4 are defined in Lemma 19.

Proof It holds that

$$
\begin{aligned}
&\sum_{t=0}^{T-1}\beta_t\left(\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2+\frac{1}{2m}\left(\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2+\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2\right)\right)\\
&\overset{(27)}{\leq}\sum_{t=0}^{T-1}\beta_t\left(\frac{8L^2\gamma^2}{m}\left\|\mathbf{\Pi}\mathbf{x}_t\right\|^2+\frac{L\gamma}{m\tau}\left\|\mathbf{\Pi}\mathbf{y}_t\right\|^2+\frac{2}{m}\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2+\frac{1}{16m}\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_t\right\|^2\right)\\
&+\sum_{t=0}^{T-1}\beta_t\left(8L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+8L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&\leq\frac{4}{m}\sum_{t=0}^{T-1}\max\left\{8L^2\gamma^2,\frac{L\gamma}{\tau c_1},\frac{2}{c_2},\frac{1}{16c_3}\right\}\cdot\beta_t\left(\left\|\mathbf{\Pi}\mathbf{x}_t\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_t\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_t\right\|^2\right)\\
&+\sum_{t=0}^{T-1}\beta_t\left(8L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+8L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&\overset{(28)(33)}{\leq}\frac{4c_5}{m}\sum_{t=0}^{T-1}\beta_t\left(\frac{1}{2}\right)^t\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right)\\
&+8c_4c_5\sum_{t=0}^{T-2}\beta_t\left(L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&+\sum_{t=0}^{T-1}\beta_t\left(8L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+8L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&\leq\frac{4c_5}{m}\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right)\cdot\sum_{t=0}^{T-1}\beta_t2^{-t}\\
&+8(1+c_4c_5)\sum_{t=0}^{T-2}\beta_t\left(L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&+\beta_{T-1}\left(8L\gamma^2D_f(\overline{y}_{T-1},\mathbf{x}_T)+8L^2\gamma^2\left\|\overline{x}_{T-1}-\overline{y}_{T-1}\right\|^2\right),
\end{aligned}
$$

where the last inequality is because of Lemma 23 with $\Delta_t=\left\|\mathbf{\Pi}\mathbf{x}_t\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_t\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_t\right\|^2$, $\tilde{\Delta}_t=c_4\left(mL\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+mL^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)$ and $\rho=1/2$. ■

4.3 Proof of Theorem 2

Combining results in previous sections, we can prove Theorem 2.

Proof First, by the setting of the step size and the definition τ, we have

$$
L^2\gamma^2=\frac{L}{400\mu}\leq\frac{1}{32\cdot73\cdot\mu^2\cdot\frac{1}{400\mu L}}=\frac{1}{32\cdot73\cdot\tau^2}.
$$

Thus, the condition $L^2\gamma^2 \leq \frac{1}{32\cdot 73\cdot \tau^2}$ in Lemma 19 holds. Furthermore, by the setting of K in Theorem 2, the results in Lemma 19 hold.

By the definition of c_1, c_2, c_3, c_4 and c_5 which are defined in Lemma 19 and Lemma 21 respectively, we can obtain that

$$c_5 = \max\left\{8L^2\gamma^2, \frac{L\gamma}{\tau c_1}, \frac{2}{c_2}, \frac{1}{16c_3}\right\} = \frac{2}{c_2} = \frac{1}{36\tau^2}.$$

This leads to $8(1 + c_4c_5) \leq 46$. Thus, the result of Lemma 21 can be represented as

$$\begin{aligned}
&\sum_{t=0}^{T-1} \beta_t \left(\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 + \frac{1}{2m}\left(\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + \left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2\right)\right) \\
&\leq \frac{1}{9m\tau^2}\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2 + c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2 + c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2 + c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right)\cdot\sum_{t=0}^{T-1}\beta_t 2^{-t} \\
&+ 46\sum_{t=0}^{T-2}\beta_t\left(L\gamma^2 D_f(\overline{y}_t, \mathbf{x}_{t+1}) + L^2\gamma^2\left\|\overline{x}_t - \overline{y}_t\right\|^2\right) \\
&+ \beta_{T-1}\left(8L\gamma^2 D_f(\overline{y}_{T-1}, \mathbf{x}_T) + 8L^2\gamma^2\left\|\overline{x}_{T-1} - \overline{y}_{T-1}\right\|^2\right).
\end{aligned} \tag{29}$$

Next, we reformulate the inequality in Lemma 15 as follows,

$$\begin{aligned}
&\frac{\gamma}{\tau}\big(F(\overline{y}_{t+1}) - F(x^*)\big) + \frac{1+\mu\gamma}{2m}\left\|\mathbf{z}_{t+1} - \mathbf{1}x^*\right\|^2 \\
&\leq \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_t) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_t - \mathbf{1}x^*\right\|^2 + \left(\frac{L\gamma}{2\tau} - \frac{1}{2\tau^2}\right)\left\|\overline{x}_{t+1} - \overline{y}_{t+1}\right\|^2 \\
&- \frac{\gamma(1-\tau)}{\tau}D_f(\overline{y}_t, \mathbf{x}_{t+1}) + \frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 + \frac{1}{2m}\left(\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + 3\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2 + 2\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2\right).
\end{aligned}$$

By the choice of γ and τ, we have

$$\frac{\gamma}{\tau}\left(\frac{\gamma(1-\tau)}{\tau}\right)^{-1} = \frac{1}{1-\tau} \geq 1 + \tau = 1 + \mu\gamma.$$

Therefore, we can extract a factor of $(1 + \mu\gamma)$ from the left.

$$\begin{aligned}
&(1+\mu\gamma)\left(\frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_{t+1}) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_{t+1} - \mathbf{1}x^*\right\|^2\right) \\
&\leq \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_t) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_t - \mathbf{1}x^*\right\|^2 + \left(\frac{L\gamma}{2\tau} - \frac{1}{2\tau^2}\right)\left\|\overline{x}_{t+1} - \overline{y}_{t+1}\right\|^2 \\
&- \frac{\gamma(1-\tau)}{\tau}D_f(\overline{y}_t, \mathbf{x}_{t+1}) + \frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2 + \frac{1}{2m}\left(\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2 + 3\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2 + 2\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2\right).
\end{aligned}$$

Multiplying this inequality by $(1+\mu\gamma)^k$, summing over iterations $k=0$ to $k=T-1$, we can obtain the bound

$$
\begin{aligned}
&(1+\mu\gamma)^T\left(\frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_T)-F(x^*)\big)+\frac{1}{2m}\left\|\mathbf{z}_T-\mathbf{1}x^*\right\|^2\right)\\
&\leq \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_0)-F(x^*)\big)+\frac{1}{2m}\left\|\mathbf{z}_0-\mathbf{1}x^*\right\|^2\\
&+\sum_{t=0}^{T-1}(1+\mu\gamma)^t\left(\left(\frac{L\gamma}{2\tau}-\frac{1}{2\tau^2}\right)\left\|\overline{x}_{t+1}-\overline{y}_{t+1}\right\|^2-\frac{\gamma(1-\tau)}{\tau}D_f(\overline{y}_t,\mathbf{x}_{t+1})\right)\\
&+\sum_{t=0}^{T-1}(1+\mu\gamma)^t\left(\frac{L\gamma}{2m\tau}\left\|\mathbf{\Pi}\mathbf{x}_{t+1}\right\|^2+\frac{1}{2m}\left(\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_{t+1}\right\|^2+3\left\|\mathbf{\Pi}\mathbf{z}_{t+1}\right\|^2+2\left\|\mathbf{\Pi}\mathbf{z}_t\right\|^2\right)\right)\\
&\overset{(29)}{\leq} \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_0)-F(x^*)\big)+\frac{1}{2m}\left\|\mathbf{z}_0-\mathbf{1}x^*\right\|^2\\
&+\sum_{t=0}^{T-1}(1+\mu\gamma)^t\left(\left(\frac{L\gamma}{2\tau}-\frac{1}{2\tau^2}\right)\left\|\overline{x}_{t+1}-\overline{y}_{t+1}\right\|^2-\frac{\gamma(1-\tau)}{\tau}D_f(\overline{y}_t,\mathbf{x}_{t+1})\right)\\
&+\frac{1}{9m\tau^2}\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right)\cdot\sum_{t=0}^{T-1}(1+\mu\gamma)^t\cdot 2^{-t}\\
&+46\sum_{t=0}^{T-2}(1+\mu\gamma)^t\left(L\gamma^2D_f(\overline{y}_t,\mathbf{x}_{t+1})+L^2\gamma^2\left\|\overline{x}_t-\overline{y}_t\right\|^2\right)\\
&+(1+\mu\gamma)^{T-1}\left(8L\gamma^2D_f(\overline{y}_{T-1},\mathbf{x}_T)+8L^2\gamma^2\left\|\overline{x}_{T-1}-\overline{y}_{T-1}\right\|^2\right)\\
&\leq \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_0)-F(x^*)\big)+\frac{1}{2m}\left\|\mathbf{z}_0-\mathbf{1}x^*\right\|^2\\
&+\sum_{t=0}^{T-2}(1+\mu\gamma)^t\left(\left(46L^2\gamma^2+\frac{L\gamma}{2\tau}-\frac{1}{2\tau^2}\right)\cdot\left\|\overline{x}_{t+1}-\overline{y}_{t+1}\right\|^2+\left(46L\gamma^2-\frac{\gamma(1-\tau)}{\tau}\right)\cdot D_f(\overline{y}_t,\mathbf{x}_{t+1})\right)\\
&+(1+\mu\gamma)^{T-1}\left(\left(8L^2\gamma^2+\frac{L\gamma}{2\tau}-\frac{1}{2\tau^2}\right)\cdot\left\|\overline{x}_T-\overline{y}_T\right\|^2+\left(8L\gamma^2-\frac{\gamma(1-\tau)}{\tau}\right)\cdot D_f(\overline{y}_{T-1},\mathbf{x}_T)\right)\\
&+\frac{1}{9m\tau^2}\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right)\cdot\sum_{t=0}^{T-1}(1+\mu\gamma)^t\cdot 2^{-t}\\
&\overset{(34)}{\leq} \frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_0)-F(x^*)\big)+\frac{1}{2m}\left\|\mathbf{z}_0-\mathbf{1}x^*\right\|^2\\
&+\frac{2}{9m\tau^2}\cdot\frac{1}{1-\mu\gamma}\cdot\left(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2+c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2+c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2+c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\right),
\end{aligned}
$$

where the last inequality is also because of the properly chosen $\gamma = \frac{1}{20\sqrt{\mu L}}$ and $\tau = \mu\gamma$ which implies that

$$46L^2\gamma^2 + \frac{L\gamma}{2\tau} - \frac{1}{2\tau^2} = \left(\frac{46}{400} + \frac{1}{2} - \frac{400}{2}\right)\frac{L}{\mu} < 0$$
$$46L\gamma^2 - \frac{\gamma(1-\tau)}{\tau} = \frac{46}{400}\cdot\frac{1}{\mu} - \frac{1}{\mu} + \frac{1}{20\sqrt{\mu L}} \overset{\mu\leq L}{\leq} \left(\frac{46}{400} + \frac{1}{20} - 1\right)\frac{1}{\mu} < 0.$$

Thus, we can obtain that

$$\begin{aligned}
&\frac{1}{2m}\left\|\mathbf{z}_T - \mathbf{1}x^*\right\|^2 \\
\leq&\frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_T) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_T - \mathbf{1}x^*\right\|^2 \\
\leq&(1+\mu\gamma)^{-T}\cdot\Big(\frac{\gamma(1-\tau)}{\tau}\big(F(\overline{y}_0) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_0 - \mathbf{1}x^*\right\|^2 \\
+&\frac{2}{9m\tau^2}\cdot\frac{1}{1-\mu\gamma}\cdot\Big(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2 + c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2 + c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2 + c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\Big)\Big) \\
\leq&\left(1 + \frac{1}{20}\cdot\sqrt{\frac{\mu}{L}}\right)^{-T}\cdot\Bigg(\frac{1}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_0 - \mathbf{1}x^*\right\|^2 \\
+&\frac{20^2L}{4m\mu}\cdot\Big(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2 + c_1\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2 + c_2\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2 + c_3\gamma^2\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\Big)\Bigg) \\
=&\left(1 + \frac{1}{20}\cdot\sqrt{\frac{\mu}{L}}\right)^{-T}\cdot\Bigg(\frac{1}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \frac{1}{2m}\left\|\mathbf{z}_0 - \mathbf{1}x^*\right\|^2 \\
+&\frac{20^2L}{4m\mu}\cdot\Big(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2 + 8\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2 + \frac{9\mu}{50L}\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2 + \frac{3}{20^4\cdot L^2}\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\Big)\Bigg)
\end{aligned}$$

Furthermore, by the fact that it holds that $(1+a)^{-1} \leq 1 - \frac{a}{2}$ if $0 \leq a \leq 1$ and multiply $2m$ both sides of above equation, we can obtain that

$$\begin{aligned}
\left\|\mathbf{z}_T - \mathbf{1}x^*\right\|^2 \leq&\left(1 - \frac{1}{40}\cdot\sqrt{\frac{\mu}{L}}\right)^{-T}\cdot\Bigg(\frac{2m}{\mu}\big(F(\overline{y}_0) - F(x^*)\big) + \left\|\mathbf{z}_0 - \mathbf{1}x^*\right\|^2 \\
&+\frac{20^2L}{2\mu}\cdot\Big(\left\|\mathbf{\Pi}\mathbf{x}_0\right\|^2 + 8\left\|\mathbf{\Pi}\mathbf{y}_0\right\|^2 + \frac{9\mu}{50L}\left\|\mathbf{\Pi}\mathbf{z}_0\right\|^2 + \frac{3}{20^4\cdot L^2}\left\|\mathbf{\Pi}\mathbf{s}_0\right\|^2\Big)\Bigg).
\end{aligned}$$

■

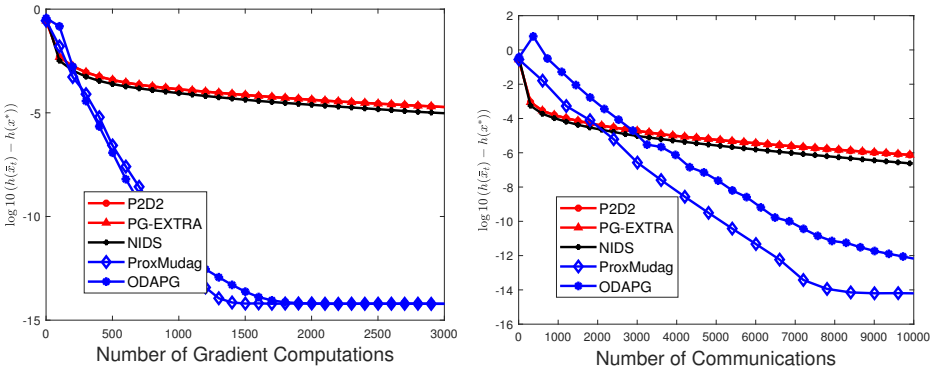
5. Experiments

We have provided a comprehensive theoretical analysis of our algorithm in the preceding sections. This section is dedicated to the empirical validation of our algorithm's effectiveness

and computational efficiency. Our experiments will focus on the sparse logistic regression problem, whose objective function adheres to the form delineated in (1), characterized by:

$$f_i(x) = \frac{1}{n}\sum_{j=1}^{n} \log\left(1+\exp(-b_{ij}\langle a_{ij}, x\rangle)\right), \qquad \text{and} \qquad g(x) = \sigma\|x\|_1 + \frac{\mu}{2}\|x\|^2, \tag{30}$$

where $a_{ij} \in \mathbb{R}^d$ and $b_{ij} \in \{-1, 1\}$ are the j-th input vector and the corresponding label on the i-th agent. It can be observed that each $f_i(x)$ in Eq. (30) exhibits both convexity and smoothness. Simultaneously, $g(x)$ demonstrates μ-strong convexity. Consequently, the sparse logistic regression problem fulfills the assumptions requisite for our algorithm.

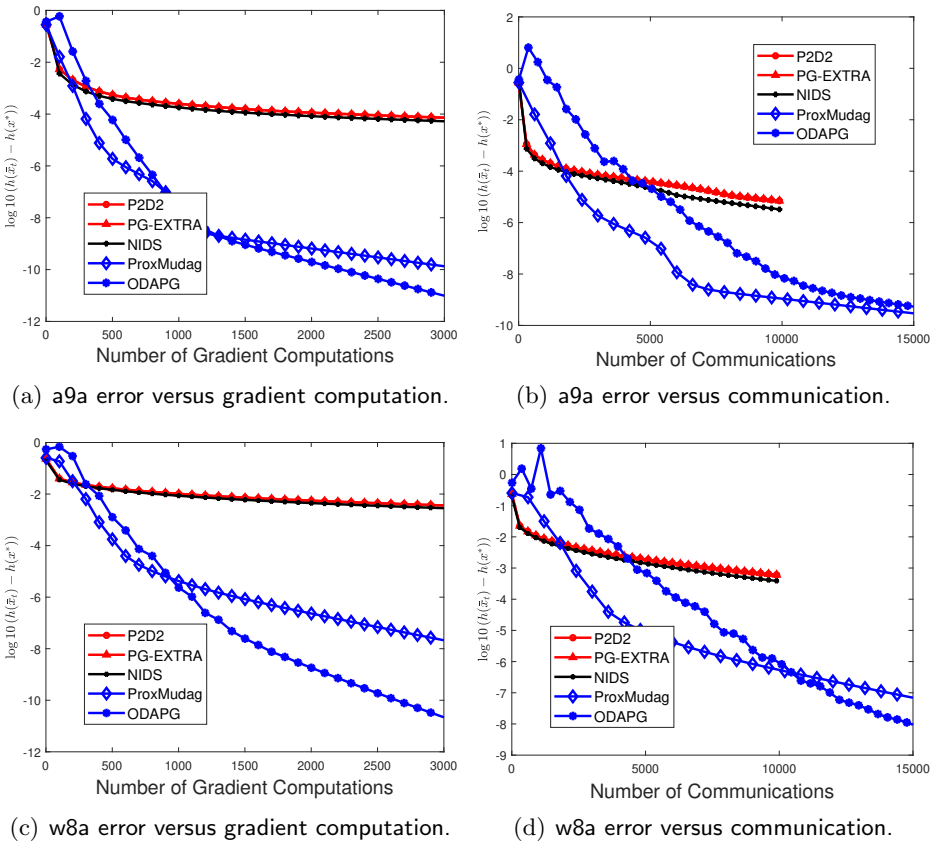


(a) a9a error versus gradient computation. (b) a9a error versus communication.

(c) w8a error versus gradient computation. (d) w8a error versus communication.

Figure 1: Algorithm evaluation with comparison to the decentralized optimization methods on the network with $1 - \lambda_2(W) = 0.05$ and $\mu = 10^{-4}$.

Experiments Setting In our experiments, we consider random networks where each pair of agents has a connection with a probability of $p = 0.1$. We set $W = I - \frac{\mathbf{L}}{\lambda_1(\mathbf{L})}$ where $\mathbf{L}$ is the Laplacian matrix associated with a weighted graph, and $\lambda_1(\mathbf{L})$ is the largest eigenvalue of $\mathbf{L}$. We set $m = 100$, that is, there are 100 agents in this network. The gossip matrix W satisfies $1 - \lambda_2(W) = 0.05$ in our experiments.



(a) a9a error versus gradient computation. (b) a9a error versus communication.

(c) w8a error versus gradient computation. (d) w8a error versus communication.

Figure 2: Algorithm evaluation with comparison to the decentralized optimization methods on the network with $1 - \lambda_2(W) = 0.05$ and $\mu = 10^{-5}$.

We conduct experiments on the datasets w8a' and a9a', which can be downloaded from the libsvm datasets. For w8a', we set $n = 497$ and $d = 300$. For a9a', we set $n = 325$ and $d = 123$. We set $\sigma = 10^{-4}$ for all datasets and choose $\mu = 10^{-4}$ and $\mu = 10^{-5}$ respectively, leading to a large condition number for the objective function.

Comparison with Existing Works We compare our work with state-of-the-art algorithms `PG-EXTRA` (Shi et al., 2015a), `NIDS` (Li et al., 2019), Decentralized Proximal Algorithm (`DPA`) (Alghunaim et al., 2019) and `ProxMudag` (Ye et al., 2023). Both our algorithm and `ProxMudag` need "FastMix" to achieve consensus. In our experiments, we set $K = 3$ in the "FastMix" for our algorithm and `ProxMudag`. In our experiments, the parameters of all algorithms are well-tuned. We report experiment results in Figure 1-2. We can observe that `ODAPG` and `ProxMudag` take much less computational cost than other algorithms because these two algorithms use Nesterov's acceleration to achieve faster convergence rates. That is, the computation complexity of `ODAPG` and `ProxMudag` is linear to $\sqrt{L/\mu}$ while the computation complexities of other algorithms without the acceleration, such as `NIDS`, are linear to L/μ. This matches our theoretical analysis of `ODAPG` over the computation com-

plexity. ODAPG also shows great advantages over other decentralized proximal algorithms without the acceleration of the communication cost. Though ODAPG takes three times of local communication while other algorithms communicate only once for each iteration, ODAPG still requires much less communication costs because of its fast convergence rate. Compared with ProxMudag, ODAPG achieves a similar computation cost to that of ProxMudag. This is because both these two algorithms apply Nesterov's acceleration to promote the convergence rates. For the communication cost, our ODAPG takes a little more communication cost to achieve the same precision solution for $\mu = 10^{-4}$. This is because ODAPG takes three "FastMix" steps while ProxMudag takes two "FastMix" steps for each iteration of the algorithms. However, when $\mu = 10^{-5}$, the value $\frac{L}{\mu}$ is larger than the case $\mu = 10^{-4}$, we can observe that our ODAPG can achieve lower communication complexities than ProxMudag on the "w8a" dataset. This empirically validates that our ODAPG can achieve a communication complexity $\mathcal{O}\left(\log \frac{L}{\mu}\right)$ lower than that of ProxMudag.

6. Conclusion and Future Work

In this paper, we have proposed the inaugural optimal algorithm for the decentralized composite optimization problem, characterized by a computational complexity of $\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\epsilon}\right)$ and a communication complexity of $\mathcal{O}\left(\sqrt{\frac{L}{\mu(1-\lambda_2(W))}}\right)$. Our algorithm accomplishes the optimal computational and communication complexities when $f(x)$ is smooth, and the regularization term $g(x)$ is strongly convex. For cases where $f_i(x)$ is L-smooth and μ-strongly convex, while $g(x)$ is convex, our algorithm can be readily adapted to address these scenarios, also achieving optimal computational and communication complexities. The effectiveness and efficiency of our algorithm, in both computation and communication, are corroborated through our experiments.

We outline two promising directions for future research.

- **Single-loop communication without multi-consensus.** Our current method employs multi-consensus (an inner communication loop) to suppress consensus errors. An important next step is to eliminate this inner loop and design a single-loop variant that preserves the optimal computation and communication complexities. In particular, leveraging the single-loop communication framework developed for smooth objectives in Song et al. (2023) and integrating loopless acceleration/consensus filtering so that ODAPG attains the same optimal rates without multi-consensus.

- **Randomized/finite-sum extensions with composite regularization.** Another natural avenue is to develop variance-reduced decentralized composite algorithms (e.g., incorporating Katyusha/SARAH/PAGE-type estimators into our proximal framework) for finite-sum problems and to study both static and time-varying networks. This line builds on recent progress in decentralized variance reduction for static graphs (Hendrikx et al., 2021; Li et al., 2020) and for time-varying graphs (Metelev et al., 2024).

Acknowledgments

Haishan Ye's work was supported by the National Key Research and Development Project of China under Grant 2022YFA1004002 and National Natural Science Foundation of China under Grant 72471185. Xiangyu Chang's work was supported by the MOE Project of Key Research Institute of Humanities and Social Sciences (22JJD110001) and National Natural Science Foundation of China under Grant 12326615.

Appendix A. Some Useful Lemmas

Lemma 22 *Letting x and y be two d-dimensional vectors, then it holds that*

$$\langle x, y\rangle \le \|x\| \|y\| \le \frac{\|x\|^2 + \|y\|^2}{2}, \tag{31}$$

and

$$\|x+y\|^2 \le 2\left(\|x\|^2 + \|y\|^2\right). \tag{32}$$

Lemma 23 *If $\Delta_{t+1} \le (1-\rho)\Delta_t + \tilde{\Delta}_t$, where Δ_t and $\tilde{\Delta}_t$ are non-negative for $t = 0, 1, \ldots$ and $\rho \in (0,1)$, given a non-negative sequence $\{\beta_t\}$, then it holds that*

$$\sum_{t=0}^{T-1} \beta_t \left((1-\rho)\cdot\Delta_t + \tilde{\Delta}_t\right) \le \sum_{k=0}^{T-1}(1-\rho)^{k+1}\Delta_0 + \frac{1}{\rho}\sum_{k=0}^{T-1}\beta_t\tilde{\Delta}_k. \tag{33}$$

Proof It holds that

$$\begin{aligned}\sum_{t=0}^{T-1} \beta_t \left((1-\rho)\cdot\Delta_t + \tilde{\Delta}_t\right) &\le \sum_{t=0}^{T-1} \beta_t \left((1-\rho)^{t+1}\Delta_0 + \sum_{\ell=1}^{t}(1-\rho)^{t-\ell}\tilde{\Delta}_\ell\right)\\ &\le \sum_{t=0}^{T-1}\beta_t(1-\rho)^{t+1}\Delta_0 + \frac{1}{\rho}\sum_{t=0}^{T-1}\beta_t\tilde{\Delta}_t,\end{aligned}$$

where the last inequality is because of $\sum_{\ell=1}^{t}(1-\rho)^{t-1} \le \frac{1}{1-(1-\rho)} = \frac{1}{\rho}$. ■

Lemma 24 *If $0 < \mu\gamma < 1$, given $T > 1$, then it holds that*

$$\sum_{t=0}^{T-1}(1+\mu\gamma)^t \cdot 2^{-t} \le \frac{2}{1-\mu\gamma}. \tag{34}$$

Proof It holds that

$$\sum_{t=0}^{T-1}(1+\mu\gamma)^t \cdot 2^{-t} = \frac{1-(1+\mu\gamma)^T\cdot 2^{-T}}{1-\frac{1+\mu\gamma}{2}} \le \frac{2}{1-\mu\gamma}.$$

■

Lemma 25 (Lemma 3 of Driggs et al. (2022)) *Suppose g is μ-strongly convex with $\mu \ge 0$, and $z = \mathrm{prox}_{\gamma g}(x - \gamma d)$ for some $x, d \in \mathbb{R}^d$ and constant γ. Then for $y \in \mathbb{R}^d$, it holds that*

$$\eta\langle d, z-y\rangle \le \frac{1}{2}\|x-y\|^2 - \frac{1+\mu\eta}{2}\|z-y\|^2 - \frac{1}{2}\|z-x\| - \eta g(z) + \eta g(y). \tag{35}$$

Lemma 26 *If $f_i(\cdot)$ is convex and L-smooth, then it holds that for $x, y \in \mathbb{R}^d$*

$$\|\nabla f_i(x) - \nabla f_i(y)\|^2 \le 2L\big(f_i(x) - f_i(y) - \langle \nabla f_i(y), x-y\rangle\big), \tag{36}$$

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L\|x-y\|. \tag{37}$$

References

Sulaiman A. Alghunaim, Kun Yuan, and Ali H. Sayed. A linearly convergent proximal gradient algorithm for decentralized optimization. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 2844–2854, 2019.

Sulaiman A Alghunaim, Ernest K Ryu, Kun Yuan, and Ali H Sayed. Decentralized proximal gradient algorithms with linear convergence rates. *IEEE Transactions on Automatic Control*, 66(6):2787–2794, 2020.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Paolo Di Lorenzo and Gesualdo Scutari. Next: In-network nonconvex optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2(2):120–136, 2016.

Alejandro D Dominguez-Garcia, Stanton T Cady, and Christoforos N Hadjicostis. Decentralized optimal dispatch of distributed energy resources. In *2012 IEEE 51st IEEE conference on decision and control (CDC)*, pages 3688–3693. IEEE, 2012.

Derek Driggs, Matthias J Ehrhardt, and Carola-Bibiane Schönlieb. Accelerating variance-reduced stochastic gradient methods. *Mathematical Programming*, pages 1–45, 2022.

Hadrien Hendrikx, Francis Bach, and Laurent Massoulie. An optimal algorithm for decentralized finite-sum optimization. *SIAM Journal on Optimization*, 31(4):2753–2783, 2021.

Dmitry Kovalev, Adil Salim, and Peter Richtárik. Optimal and practical algorithms for smooth and strongly convex decentralized optimization. *Advances in Neural Information Processing Systems*, 33:18342–18352, 2020.

Dmitry Kovalev, Elnur Gasanov, Alexander Gasnikov, and Peter Richtarik. Lower bounds and optimal algorithms for smooth and strongly convex decentralized optimization over time-varying networks. *Advances in Neural Information Processing Systems*, 34:22325–22335, 2021.

Dmitry Kovalev, Aleksandr Beznosikov, Abdurakhmon Sadiev, Michael Persiianov, Peter Richtárik, and Alexander Gasnikov. Optimal algorithms for decentralized stochastic variational inequalities. *Advances in Neural Information Processing Systems*, 35:31073–31088, 2022.

Huan Li and Zhouchen Lin. Revisiting extra for smooth distributed optimization. *SIAM Journal on Optimization*, 30(3):1795–1821, 2020.

Huan Li and Zhouchen Lin. Accelerated gradient tracking over time-varying graphs for decentralized optimization. *arXiv preprint arXiv:2104.02596*, 2021.

Huan Li, Zhouchen Lin, and Yongchun Fang. Optimal accelerated variance reduced extra and diging for strongly convex and smooth decentralized optimization. *arXiv preprint arXiv:2009.04373*, 79(85):140, 2020.

Zhi Li, Wei Shi, and Ming Yan. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Transactions on Signal Processing*, 67(17):4494–4506, 2019.

Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. *Advances in neural information processing systems*, 30, 2017.

Ji Liu and A. Stephen Morse. Accelerated linear iterations for distributed averaging. *Annual Reviews in Control*, 35(2):160–165, 2011.

Marie Maros and Joakim Jaldén. Panda: A dual linearly converging method for distributed optimization over time-varying undirected graphs. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 6520–6525. IEEE, 2018.

Dmitry Metelev, Savelii Chezhegov, Alexander Rogozin, Aleksandr Beznosikov, Alexander Sholokhov, Alexander Gasnikov, and Dmitry Kovalev. Decentralized finite-sum optimization over time-varying networks. *arXiv preprint arXiv:2402.02490*, 2024.

Aryan Mokhtari, Wei Shi, Qing Ling, and Alejandro Ribeiro. A decentralized second-order method with exact linear convergence rate for consensus optimization. *IEEE Trans. Signal Inf. Process. over Networks*, 2(4):507–522, 2016.

Angelia Nedic, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4): 2597–2633, 2017.

Yu Nesterov. Gradient methods for minimizing composite functions. *Mathematical programming*, 140(1):125–161, 2013.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International conference on machine learning*, pages 2613–2621. PMLR, 2017.

Shi Pu, Wei Shi, Jinming Xu, and Angelia Nedić. Push–pull gradient methods for distributed optimization in networks. *IEEE Transactions on Automatic Control*, 66(1): 1–16, 2020.

Guannan Qu and Na Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2017.

Guannan Qu and Na Li. Accelerated distributed nesterov gradient descent. *IEEE Transactions on Automatic Control*, 65(6):2566–2581, 2019.

Ali H Sayed et al. Adaptation, learning, and optimization over networks. *Foundations and Trends® in Machine Learning*, 7(4-5):311–801, 2014.

Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *international conference on machine learning*, pages 3027–3036. PMLR, 2017.

Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Trans. Signal Process.*, 63(22):6013–6023, 2015a.

Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015b.

Zhuoqing Song, Lei Shi, Shi Pu, and Ming Yan. Optimal gradient tracking for decentralized optimization. *Mathematical Programming*, pages 1–53, 2023.

Ying Sun, Gesualdo Scutari, and Amir Daneshmand. Distributed optimization based on gradient tracking revisited: Enhancing convergence rate via surrogation. *SIAM Journal on Optimization*, 32(2):354–385, 2022.

Tong Tong Wu, Yi Fang Chen, Trevor Hastie, Eric Sobel, and Kenneth Lange. Genome-wide association analysis by lasso penalized logistic regression. *Bioinformatics*, 25(6): 714–721, 2009.

Jinming Xu, Ye Tian, Ying Sun, and Gesualdo Scutari. Distributed algorithms for composite optimization: Unified framework and convergence analysis. *IEEE Transactions on Signal Processing*, 69:3555–3570, 2021.

Haishan Ye, Luo Luo, Ziang Zhou, and Tong Zhang. Multi-consensus decentralized accelerated gradient descent. *Journal of Machine Learning Research*, 24(306):1–50, 2023.

Kun Yuan, Bicheng Ying, Xiaochuan Zhao, and Ali H. Sayed. Exact diffusion for distributed optimization and learning - part I: algorithm development. *IEEE Trans. Signal Process.*, 67(3):708–723, 2019.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 2005.