A Unifying Approximate Method of Multipliers for Distributed Composite Optimization

Xuyang Wu and Jie Lu, Member, IEEE

Abstract—This article investigates solving convex composite optimization on an undirected network, where each node, privately endowed with a smooth component function and a nonsmooth one, is required to minimize the sum of all the component functions throughout the network. To address such a problem, a general approximate method of multipliers (AMM) is developed, which attempts to approximate the method of multipliers by virtue of a surrogate function with numerous options. We then design the possibly nonseparable, time-varying surrogate function in various ways, leading to different distributed realizations of AMM. We demonstrate that AMM generalizes more than ten state-of-the-art distributed optimization algorithms, and certain specific designs of its surrogate function result in a variety of new algorithms to the literature. Furthermore, we show that AMM is able to achieve an $O(1/k)$ rate of convergence to optimality, and the convergence rate becomes linear when the problem is locally restricted strongly convex and smooth. Such convergence rates provide new or stronger convergence results to many prior methods that can be viewed as specializations of AMM.

Index Terms—Convex composite optimization, distributed optimization, method of multipliers.

I. INTRODUCTION

This article addresses the following convex composite optimization problem:

$$\min_{x \in \mathbb{R}^d} \sum_{i=1}^{N} (f_i(x) + h_i(x))$$  \hspace{1cm} (1)

where each $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ is convex and has a Lipschitz continuous gradient, and each $h_i : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ is convex and can be nondifferentiable. Note that $f_i$ is allowed to be a zero function. Also, if $h_i$ contains an indicator function $\mathcal{I}_{X_i}$ with respect to a closed convex set $X_i \subset \mathbb{R}^d$, i.e., $\mathcal{I}_{X_i}(x) = 0$ if $x \in X_i$ and $\mathcal{I}_{X_i}(x) = +\infty$ otherwise, then problem (1) turns into a nonsmooth, constrained convex program.

We consider solving problem (1) in a distributed way over a network modeled as a connected, undirected graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, N\}$ is the vertex set and $\mathcal{E} \subseteq \{(i, j) | i, j \in \mathcal{V}, i \neq j\}$ is the edge set. We suppose each node $i \in \mathcal{V}$ possesses two private component functions $f_i$ and $h_i$, aims at solving (1), and only exchanges information with its neighbors denoted by the set $\mathcal{N}_i = \{j | \{i, j\} \in \mathcal{E}\}$. Applications of such a distributed composite optimization problem include control of multi-agent systems, robust estimation by sensor networks, resource allocation in smart grids, decision making for swarm robotics, as well as distributed learning [1].

To date, a large body of distributed optimization algorithms have been proposed to solve problem (1) or its special cases. A typical practice is to reformulate (1) into a constrained problem with a separable objective function and a consensus constraint to be relaxed. For example, the inexact methods [2], [3] solve this reformulated problem by penalizing the consensus constraint in the objective function, and the primal-dual methods [4]–[22] relax the consensus constraint through dual decomposition techniques. Another common approach is to emulate the centralized subgradient/gradient methods in a decentralized manner, such as the distributed subgradient methods [23], [24] and the distributed gradient-tracking-based methods [25]–[31], where [28]–[31] utilize surrogate functions in order to realize the decentralized emulation.

Despite the growing literature, relatively few methods manage to tackle the general form of problem (1) under distributed settings, among which only the first-order methods [6]–[13], [15]–[17] are guaranteed to converge to optimality with constant step-sizes. Here, [8] and [15] rely on strong convexity of the problem, and [6], [10], and [17] only prove asymptotic convergence for non-strongly convex problems. In contrast, the remaining works [7], [9], [11]–[13], and [16] establish $O(1/k)$ convergence rates in solving (1). Although these algorithms are developed using different rationales, we discover that the majority of them, i.e., [9], [11]–[13], [16], can indeed be thought to originate from the method of multipliers [32].

The method of multipliers is a seminal (centralized) optimization method, and one of its notable variants is the alternating direction method of multipliers (ADMM) [32]. In this article, we develop a novel paradigm of solving (1) via approximating the behavior of the method of multipliers, called approximate method of multipliers (AMM). The proposed AMM adopts a possibly time-varying surrogate function to take the place...
of the smooth objective function at every minimization step in the method of multipliers, facilitating abundant designs of distributed algorithms for solving (1) in a fully decentralized fashion. Unlike the typical separable surrogate functions for distributed optimization such as those in [28]–[31], our surrogate function can be nonseparable in some distributed versions of AMM. It also admits certain function forms that cannot be included by [28]–[31].

To enable distributed implementation of AMM, we first opt for a Bregman-divergence-type surrogate function, leading to a class of distributed realizations of AMM. We also utilize convex conjugate and graph topologies to design the surrogate function and, thus, construct two additional sets of distributed realizations of AMM for solving smooth convex optimization [i.e., (1) with (1)] with convexity parameter $\rho > 0$. We concretely exemplify such realizations of AMM, so that new distributed proximal/second-order/gradient-tracking methods can be obtained. Apart from that, AMM and its distributed realizations also unify a wide range of state-of-the-art distributed first-order and second-order algorithms [4], [5], [9], [11]–[13], [16], [18], [19], [25] for solving (1) or its special cases. This offers a unifying perspective for understanding the nature of these existing methods with various design rationales.

We show that AMM allows the nodes to reach a consensus and converge to the optimal value at a rate of $O(1/k)$, either with a particular surrogate function type or under a local restricted strong convexity condition on $\sum_{i \in \mathcal{V}} f_i(x)$. For all the distributed algorithms that are able to solve the general form of problem (1), the convergence rates of AMM and the algorithms in [7], [9], [11]–[13], and [16] achieve the best order of $O(1/k)$, while the algorithms in [9], [11]–[13], and [16] are indeed special cases of AMM with that particular surrogate function type and the convergence rate in [7] is in terms of an implicit measure of optimality error. Moreover, when problem (1) is smooth and locally restricted strongly convex, AMM enables the nodes to attain the optimum at a linear rate. Unlike most existing works, this linear convergence is established in need of global strong convexity. Naturally, our analysis on AMM yields new convergence results and relaxes some problem assumptions with no degeneration of the convergence rate order for quite a few existing methods that can be generalized by AMM.

The outline of the article is as follows. Section II describes the proposed AMM and the distributed designs. Section III demonstrates that AMM generalizes a number of prior methods. Section IV discusses the convergence results of AMM, and Section V presents the comparative simulation results. Finally, Section VI concludes the article.

**Notation and Definition:** We use $A = (a_{ij})_{n \times n}$ to denote an $n \times n$ real matrix whose $(i, j)$-entry, denoted by $[A]_{ij}$, is equal to $a_{ij}$. In addition, $\text{Null}(A)$, Range $(A)$, and $\|A\|$ are the null space, range, and spectral norm of $A$, respectively. Besides, $\text{diag}(D_1, \ldots, D_n) \in \mathbb{R}^{n \times nd}$ represents the block diagonal matrix with $D_1, \ldots, D_n \in \mathbb{R}^{d \times d}$ being its diagonal blocks. Also, $0$, $1$, and $\mathbf{O}$ represent a zero vector, an all-one vector, and a zero square matrix of proper dimensions, respectively, and $I_n$ represents the $n \times n$ identity matrix. For any two matrices $A$ and $B$, $A \otimes B$ is their Kronecker product. For any $A = AT \in \mathbb{R}^{n \times n}$, $\lambda_{\text{max}}(A)$ denotes the largest eigenvalue of $A$. For any $z \in \mathbb{R}^n$ and $A \succeq 0$, $\|z\| = \sqrt{z^T z}$ and $\|z\|_A = \sqrt{z^T A z}$. For any convex set $X$, $P_X[ : ]$ is the projection onto $X$. For any countable set $S$, $|S|$ represents its cardinality. For any convex function $f : \mathbb{R}^d \rightarrow \mathbb{R}$, $\partial f(x) \subset \mathbb{R}^d$ denotes its subdifferential (i.e., the set of subgradients) at $x$. If $f$ is differentiable, $\partial f(x)$ only contains the gradient $\nabla f(x)$.

Given a convex set $X \subseteq \mathbb{R}^d$, a function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be strongly convex on $X$ with convexity parameter $\rho > 0$ (or simply $\sigma$-strongly convex on $X$) if $g(x) \geq \sigma(|x-y|)^2$, $\forall x, y \in X \forall g_x \in \partial f(x) \forall g_y \in \partial f(y)$. We say $f$ is (globally) strongly convex if it is strongly convex on $\mathbb{R}^d$. Given $x, f$, $f$ is said to be restricted strongly convex with respect to $\hat{x}$ on $X$ with convexity parameter $\rho > 0$ if $g(x) \geq \sigma(|x-x|)^2$, $\forall x \in X \forall g_x \in \partial f(x) \forall g_x \in \partial f(x)$. Given $L \geq 0$, $f$ is said to be $L$-smooth if it is differentiable and its gradient is Lipschitz continuous with Lipschitz constant $L$, i.e., $\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|$.

**II. APPROXIMATE METHOD OF MULTIPLIERS AND DISTRIBUTED DESIGNS**

This section develops distributed algorithms for solving (1) over the graph $\mathcal{G}$ under Assumption 1.

**Assumption 1:** For each $i \in \mathcal{V}$, $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ and $h_i : \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$ are closed convex functions. In addition, $f_i$ is $M_i$-smooth for some $M_i > 0$. Moreover, there exists at least one optimal solution $x^*$ to problem (1).

**A. Approximate Method of Multipliers**

We first propose a family of algorithms that effectively approximate the method of multipliers [32] and serve as a cornerstone of solving (1) in a distributed way.

Let each node $i \in \mathcal{V}$ keep a copy $x_i \in \mathbb{R}^d$ of the global decision variable $x \in \mathbb{R}^d$ in problem (1), and define

$$f(x) := \sum_{i \in \mathcal{V}} f_i(x_i), \quad h(x) := \sum_{i \in \mathcal{V}} h_i(x_i)$$

where $x = (x_1, \ldots, x_N)^T \in \mathbb{R}^{Nd}$. Then, problem (1) can be equivalently transformed into

$$\begin{align*}
\text{minimize} & \quad f(x) + h(x) \\
\text{subject to} & \quad x \in S := \{x \in \mathbb{R}^{Nd} | x_1 = \cdots = x_N\}.
\end{align*}$$

Next, let $\hat{H} \in \mathbb{R}^{Nd \times Nd}$ be such that $\hat{H} = \hat{H}^T$, $\hat{H} \succeq 0$, and $\text{Null}(\hat{H}) = S$. It has been shown in [18] that the consensus constraint $x \in S$ in problem (2) is equivalent to $\hat{H} \hat{x} = 0$. Therefore, (2) is equivalent to

$$\begin{align*}
\text{minimize} & \quad f(x) + h(x) \\
\text{subject to} & \quad \hat{H} \hat{x} = 0.
\end{align*}$$

Problems (1), (2), and (3) have the same optimal value. Also, $x^* \in \mathbb{R}^d$ is an optimum of (1) if and only if $x^* = ((x_1^*)^T, (x_N^*)^T)^T \in \mathbb{R}^{Nd}$ is an optimum of (2) and (3).

Consider applying the method of multipliers [32] to solve problem (3), which gives

$$x^{k+1} = \arg \min_{x \in \mathbb{R}^{Nd}} f(x) + h(x) + \frac{\mu}{2} \|x - (x^k)^T \hat{H} \|_2^2.$$
\[ \mathbf{x}^{k+1} = \mathbf{x}^k + \rho \hat{H} \hat{\mathbf{z}} \mathbf{v}^k + 1. \\
\]

Here, \( \mathbf{x}^k \in \mathbb{R}^N \) is the primal variable at iteration \( k \geq 0 \), which is updated by minimizing an augmented Lagrangian function with the penalty \( \frac{\rho}{2} \| \mathbf{x} \|_2^2 \), \( \rho > 0 \) on the consensus constraint \( \hat{H} \hat{\mathbf{z}} \mathbf{x} = \mathbf{0} \). In addition, \( \mathbf{v}^k \in \mathbb{R}^N \) is the dual variable, whose initial value \( \mathbf{v}^0 \) can be arbitrarily set.

Although the method of multipliers may be applied to solve (3) with properly selected parameters, it is not implementable in a distributed fashion over \( G \), even when the problem reduces to linear programming. To address this issue, our strategy is to first derive a paradigm of approximating the method of multipliers and then design its distributed realizations.

Our approximation approach is as follows: Starting from any \( \mathbf{v}^0 \in \mathbb{R}^N \), let
\[
\begin{align*}
\mathbf{x}^{k+1} &= \arg \min_{\mathbf{x} \in \mathbb{R}^N} u^k(\mathbf{x}) + h(\mathbf{x}) + \frac{\rho}{2} \| \mathbf{x} \|_H^2 + (\mathbf{v}^k)^T \hat{H} \hat{\mathbf{z}} \mathbf{x}, \ (4) \\
\mathbf{v}^{k+1} &= \mathbf{v}^k + \rho \hat{H} \hat{\mathbf{z}} \mathbf{v}^k, \ \forall k \geq 0. \\
\end{align*}
\]

Compared to the method of multipliers, we adopt the same dual update (5) but construct a different primal update (4). In (4), we use a possibly time-varying surrogate function \( u^k : \mathbb{R}^N \to \mathbb{R} \) to replace \( f(\mathbf{x}) \) in the primal update of the method of multipliers, whose conditions are imposed in Assumption 2. Additionally, to introduce more flexibility, we use a different weight matrix \( H \in \mathbb{R}^{N \times N} \) to define the penalty term \( \frac{\rho}{2} \| \mathbf{x} \|_H^2 \). \( \rho > 0 \). We suppose \( H \) has the same properties as \( \hat{H} \), i.e., \( H = H^T \succ 0 \) and \( \text{Null}(H) = S \).

**Assumption 2:** The functions \( u^k \forall k \geq 0 \) satisfy the following.

1. \( u^k \forall k \geq 0 \) are convex and twice continuously differentiable.
2. \( u^k + \frac{\rho}{2} \| \cdot \|_H^2 \forall k \geq 0 \) are strongly convex, whose convexity parameters are uniformly bounded from below by some positive constant.
3. \( \nabla u^k \forall k \geq 0 \) are Lipschitz continuous, whose Lipschitz constants are uniformly bounded from above by some nonnegative constant.
4. \( \nabla u^k(\mathbf{x}^k) = \nabla f(\mathbf{x}^k) \forall k \geq 0 \), where \( \mathbf{v}^0 \) is arbitrarily given and \( \mathbf{x}^k \), \( k \geq 1 \) is generated by (4) and (5).

Assumption 2(b) guarantees that \( \mathbf{x}^{k+1} \in (4) \) is well-defined and uniquely exists. Assumption 2(d) is the key to making (4) and (5) solve problem (3). To explain this, note that (4) is equivalent to finding the unique \( \mathbf{x}^{k+1} \) satisfying
\[
-\nabla u^k(\mathbf{x}^{k+1}) - \rho H \mathbf{x}^{k-1} - \hat{H} \hat{\mathbf{z}} \mathbf{v}^k \in \partial h(\mathbf{x}^{k+1}).
\]

Let \( (\mathbf{x}^*, \mathbf{v}^*) \) be a primal-dual optimal solution pair of problem (3), which satisfies
\[
-\hat{H} \hat{\mathbf{z}} \mathbf{v}^* - \nabla f(\mathbf{x}^*) \in \partial h(\mathbf{x}^*). \\
\]

If \( (\mathbf{x}^0, \mathbf{v}^0) = (\mathbf{x}^*, \mathbf{v}^*) \), then \( \mathbf{x}^{k+1} \) has to be \( \mathbf{x}^* \) because of Assumption 2(d), \( H \mathbf{x}^* = 0, (7) \), and the uniqueness of \( \mathbf{x}^{k+1} \) in (6). It follows from (5) and \( \hat{H} \hat{\mathbf{z}} \mathbf{x} = 0 \) that \( \mathbf{v}^{k+1} = \mathbf{v}^* \). Therefore, \( (\mathbf{x}^*, \mathbf{v}^*) \) is a fixed point of (4) and (5). The remaining conditions in Assumption 2 will be used for convergence analysis later.

The paradigm described by (4) and (5) and obeying Assumption 2 is called **approximate method of multipliers** (AMM).

As there are numerous options of the surrogate function \( u^k \), AMM unifies a wealth of optimization algorithms, including a variety of existing methods (cf., Section III) and many brand new algorithms. Moreover, since Assumption 2 allows \( u^k \) to have a more favorable structure than \( f \), AMM with appropriate \( u^k \)’s may induce a prominent reduction in computational cost compared to the method of multipliers. In the sequel, we will provide various options of \( u^k \), which give rise to a series of **distributed** versions of AMM.

A number of existing works [28]–[31] also introduce surrogate functions to distributed optimization methods and can address partially nonconvex problems on time-varying networks. The advantages of AMM as well as its differences from them are highlighted as follows:

1. The algorithms carrying surrogate functions in [28]–[31] are (primal) gradient-tracking-based methods. In contrast, AMM incorporates a surrogate function into a primal-dual framework, so that AMM is inherently different from the algorithms in [28]–[31].
2. The surrogate function conditions in Assumption 2 intersect with but still differ from those in [28]–[31]. For instance, when \( f \) is twice continuously differentiable, \( u^k(\mathbf{x}) = \langle \nabla f(\mathbf{x}^k), \mathbf{x} \rangle + \frac{1}{2} \| \mathbf{x} - \mathbf{x}^k \|_2^2 + (\mathbf{v}^k)^T (\nabla^2 f(\mathbf{x}^k) + \epsilon I)(\mathbf{x} - \mathbf{x}^k) \) with \( \epsilon > 0 \) meets Assumption 2 but cannot be included by [28]–[31].
3. To enable distributed implementation, the surrogate functions in [28]–[31] need to be fully separable in the sense that they can be written as the sum of \( N \) functions with independent variables. In contrast, AMM allows \( u^k \) to be densely coupled under proper design and yet can be executed in a distributed fashion (cf., Sections II-C and III-C4), leading to more diverse algorithm design.
4. In [28]–[31], it is required that the global nonsmooth objective be accessible to every node, while (1) only assigns a local nonsmooth component \( h_i \) to each node \( i \), which is more applicable to scenarios concerning privacy and makes the analysis more challenging.

**B. Distributed Approximate Method of Multipliers**

This subsection lays out the parameter designs of AMM for distributed implementations.

We first apply the following change of variable to AMM:
\[
\mathbf{q}^k = \hat{H} \hat{\mathbf{z}} \mathbf{v}^k, \quad k \geq 0. \\
\]

Then, AMM (4) and (5) can be rewritten as
\[
\begin{align*}
\mathbf{x}^{k+1} &= \arg \min_{\mathbf{x} \in \mathbb{R}^N} u^k(\mathbf{x}) + h(\mathbf{x}) + \frac{\rho}{2} \| \mathbf{x} \|_H^2 + (\mathbf{q}^k)^T \mathbf{x}, \ (9) \\
\mathbf{q}^{k+1} &= \mathbf{q}^k + \rho H \mathbf{x}^{k+1}. \\
\end{align*}
\]

Moreover, note that Range(\( \hat{H} \hat{\mathbf{z}} \)) = Range(\( \hat{H} \)) = \( S^\perp := \{ \mathbf{x} \in \mathbb{R}^N : x_1 + \cdots + x_N = 0 \} \) where \( S^\perp \) is the orthogonal complement of \( S \) in (2). Hence, (8) requires \( \mathbf{q}^k \in S^\perp \forall k \geq 0 \), which, due to (10), can be ensured simply by the following initialization:
\[
\mathbf{q}^0 \in S^\perp. \\
\]

Therefore, (9)–(11) is an equivalent form of AMM.
Next, partition the primal variable $x^k$ and the dual variable $q^k$ in (9)-(11) as $x^k = (x^k_1, \ldots, x^k_N)^T$ and $q^k = (q^k_1, \ldots, q^k_N)^T$. Suppose each node $i \in V$ maintains $x_i^k \in \mathbb{R}^d$ and $q_i^k \in \mathbb{R}^d$. Clearly, the nodes manage to collectively meet (11) by setting, for instance, $q_i^k = 0 \forall i \in V$. Below, we discuss the selections of $u^k_i$, $H$, and $\tilde{H}$ for the sake of distributed implementations of (9) and (10).

To this end, we let $H = P \otimes I_d$, $\tilde{H} = \tilde{P} \otimes I_d$ and impose Assumption 3 on $P, \tilde{P} \in \mathbb{R}^{N \times N}$.

**Assumption 3:** The matrices $P = (p_{ij})_{N \times N}$ and $\tilde{P} = (\tilde{p}_{ij})_{N \times N}$ satisfy the following.

a) $p_{ij} = p_{ji}, \tilde{p}_{ij} = \tilde{p}_{ji}, \forall i, j \in \mathcal{E}$.

b) $p_{ii} = \tilde{p}_{ii} = 0, \forall i \in V, \forall j \notin \mathcal{N}_i \cup \{i\}$.

c) Null($P$) = Null($\tilde{P}$) = span($I$).

d) $P \succeq 0, \tilde{P} \succeq 0$.

The nodes can jointly determine $P, \tilde{P}$ under Assumption 3 without any centralized coordination. For instance, we can let each node $i \in V$ agree with every neighbor $j \in \mathcal{N}_i$ on $p_{ij}$ so that $p_{ij} = p_{ji} < 0$ and $\tilde{p}_{ij} = \tilde{p}_{ji} < 0$, and set $p_{ii} = -\sum_{j \in \mathcal{N}_i} p_{ij}$ and $\tilde{p}_{ii} = -\sum_{j \in \mathcal{N}_i} \tilde{p}_{ji}$, which directly guarantee Assumption 3(a).

Next, partition the primal variable $x^k$ and the dual variable $q^k$ in (9)-(11) as $x^k = (x^k_1, \ldots, x^k_N)^T$ and $q^k = (q^k_1, \ldots, q^k_N)^T$. Suppose each node $i \in V$ maintains $x_i^k \in \mathbb{R}^d$ and $q_i^k \in \mathbb{R}^d$. Clearly, the nodes manage to collectively meet (11) by setting, for instance, $q_i^k = 0 \forall i \in V$. Below, we discuss the selections of $u^k_i$, $H$, and $\tilde{H}$ for the sake of distributed implementations of (9) and (10).

1: **Initialization:**

2: Each node $i \in \mathcal{V}$ selects $q_i^0 \in \mathbb{R}^d$ such that $\sum_{i \in \mathcal{V}} q_i^0 = 0$ (or simply sets $q_i^0 = 0$).

3: Each node $i \in \mathcal{V}$ arbitrarily sets $x_i^0 \in \mathbb{R}^d$ and sends $x_i^0$ to every neighbor $j \in \mathcal{N}_i$.

4: For $k \geq 0$ do

5: Each node $i \in \mathcal{V}$ computes $x_i^{k+1}$ via (17).

6: Each node $i \in \mathcal{V}$ sends $x_i^{k+1}$ to every neighbor $j \in \mathcal{N}_i$.

7: Each node $i \in \mathcal{V}$ computes $q_i^{k+1}$ via (12).

8: end for

Then, we set $u^k$ as

$$u^k(x) = D_{\phi}(x, x^k) + \langle \nabla f(x), x \rangle. \quad (15)$$

With Assumption 4, (15) is sufficient to ensure Assumption 2. To see this, note from Assumptions 4(a) and (d) that $u^k(x)$ in (15) is twice continuously differentiable and convex, i.e., Assumption 2(a) holds. Also note from (14) and (15) that

$$\nabla u^k(x) = \nabla \phi_k(x) - \nabla \phi_k(x^k) + \nabla f(x). \quad (16)$$

This, along with Assumptions 4(b) and (c), guarantees Assumptions 2(b)-(d).

To see how (15) results in a distributed implementation of (9), note from (16) that (9) is equivalent to

$$0 = \nabla \phi_k(x^{k+1}) - \nabla \phi_k(x^k) + \nabla f(x^k) + g^{k+1} + \rho H x^{k+1} + q^k$$

for some $g^{k+1} \in \partial h_i(x^{k+1})$. Then, using (13) and the structure of $H$ given in Assumption 3, this can be written as

$$0 = \nabla \psi_i^k(x^{k+1}) + \psi_i^{k+1} + \psi_i - \nabla \psi_i(x^k) + \nabla f_i(x^k) + \rho \sum_{j \in \mathcal{N}_i} p_{ij} x_j^k$$

where $\psi_i^{k+1} \in \partial h_i(x^{k+1})$. In other words, (9) can be achieved by letting each node $i \in \mathcal{V}$ solve the following strongly convex optimization problem:

$$x_i^{k+1} = \operatorname{arg} \min_{x \in \mathbb{R}^d} \psi_i^k(x) + h_i(x)$$

$$+ \langle x, q_i^k - \nabla \psi_i(x^k) + \nabla f_i(x^k) + \rho \sum_{j \in \mathcal{N}_i} p_{ij} x_j^k \rangle$$

$$x_i^{k+1}$$ (17)

which can be locally carried out by node $i$.

Algorithms described by (11), (17), and (12) under Assumptions 3 and 4 constitute a set of distributed realizations of AMM, referred to as distributed approximate method of multipliers (DAMM), which can be implemented by the nodes in $\mathcal{G}$ via exchanging information with their neighbors only. Algorithm 1 describes how each node acts in DAMM.

Finally, we provide two examples of DAMM with two particular choices of $\psi_i^k$.

**Example 1:** For each $i \in \mathcal{V}$, let $\psi_i^k(x) = r_i(x - x_i^k) + \frac{\epsilon_i}{2} \|x\|_2^2 \forall k \geq 0$, where $r_i : \mathbb{R}^d \to \mathbb{R}$ can be any convex function, smooth, and twice continuously differentiable function and $\epsilon_i \geq \rho \lambda_{\max}(H)$. Then, DAMM reduces to a new distributed
proximal algorithm, with the following local update of \( x_i^k \):

\[
x_i^{k+1} = \arg \min_{x \in \mathbb{R}^d} \ r_i(x - x_i^k) + \frac{\epsilon_i}{2} \| x - x_i^k \|^2 + h_i(x)
\]

\[
+ \langle x, q_i^k - \nabla r_i(0) + \nabla f_i(x_i^k) + \rho \sum_{j \in \mathcal{N} \cup \{i\}} p_{ij} x_j^k \rangle.
\]

**Example 2:** Suppose \( f_i \) \( \forall i \in \mathcal{V} \) are twice continuously differentiable. Since \( \nabla^2 f_i(x) \preceq M_i I_d \) \( \forall x \in \mathbb{R}^d \), we can let each \( \psi_i(x) = \frac{1}{2} x^T (\nabla^2 f_i(x) + \epsilon_i I_d) x \), where \( \epsilon_i \geq \rho \min_{\mathcal{V}} (M) \). Then, the resulting DAMM is a new distributed second-order method, with the following local update of \( x_i^k \):

\[
x_i^{k+1} = \arg \min_{x \in \mathbb{R}^d} \frac{1}{2} \| x - x_i^k \|^2 + \frac{\epsilon_i}{2} \| x - x_i^k \|^2 + h_i(x)
+ \langle x, q_i^k + \nabla f_i(x_i^k) + \rho \sum_{j \in \mathcal{N} \cup \{i\}} p_{ij} x_j^k \rangle.
\]

**C. Special Case: Smooth Problem**

In this subsection, we focus on the smooth convex optimization problem \( \min_{x \in \mathbb{R}^d} \sum_{i \in \mathcal{V}} f_i(x) \), i.e., (1) with \( h_i(x) = 0 \) \( \forall i \in \mathcal{V} \), and provide additional designs of \( u^k \) in AMM, leading to a couple of variations of DAMM.

Here, we let \( u^k \) still be in the form of (15) but no longer require \( \phi^k + \frac{\rho \gamma}{2} \cdot \| x \|^2_H \) to be a separable function as in (13). Instead, we construct \( \phi^k \) based upon another function \( \gamma^k : \mathbb{R}^{Nd} \rightarrow \mathbb{R} \) under Assumption 5.

**Assumption 5:** The functions \( \gamma^k \) \( \forall k \geq 0 \) satisfy the following.

a) \( \gamma^k \forall k \geq 0 \) are twice continuously differentiable.

b) \( \gamma^k \forall k \geq 0 \) are strongly convex, whose strong convexity parameters are uniformly bounded from below by \( \gamma > 0 \).

c) \( \nabla \gamma^k \forall k \geq 0 \) are Lipschitz continuous, whose Lipschitz constants are uniformly bounded from above by \( \gamma > 0 \).

d) \( (\gamma^k)^*(\cdot) - \frac{\gamma}{2} \| \cdot \|^2_H \) \( \forall k \geq 0 \) are convex, where \( (\gamma^k)^* = \sup_{y \in \mathbb{R}^{Nd}} \langle x, y \rangle - \gamma \langle y, x \rangle \) is the convex conjugate function of \( \gamma^k \).

e) For any \( k \geq 0 \) and any \( x \in \mathbb{R}^{Nd} \), the \( i \)th \( d \)-dimensional block of \( \nabla \gamma^k(x) \), denoted by \( \nabla_{x_i} \gamma^k(x) \), is independent of \( x_j \forall j \notin N_i \cup \{i\} \).

From Assumptions 5(b) and (c), each \( (\gamma^k)^*(z) \) is \((1/\gamma)\)-strongly convex and \((1/\gamma)\)-smooth [33], so that Assumption 5(d) holds as long as \( I_{Nd}/\gamma - \rho H \succeq 0 \). Now we set

\[
\phi^k(x) = (\gamma^k)^*(x) - \frac{\rho \gamma}{2} \| x \|^2_H, \quad \forall k \geq 0.
\]

Unlike DAMM, \( u^k \) given by (15) and (18) under Assumption 5 does not necessarily guarantee that \( u^k(x) + \rho \gamma \| x \|^2_H \) is separable. Below, we show that such \( u^k \forall k \geq 0 \) also satisfy Assumption 2, leading to another subclass of AMM.

To do so, first note that the strong convexity and smoothness of \( (\gamma^k)^* \) guarantee Assumptions 2(b) and (c). Also note from (16) that Assumption 2(d) is assured. In addition, due to Assumption 5(d), \( \phi^k \) is convex, and, thus, so is \( u^k \). To show the twice continuous differentiability of \( u^k \) under Assumption 2(a), consider the fact from [34] that due to Assumptions 5(b) and (c), \( \nabla \gamma^k \) is invertible and its inverse function is \( (\nabla \gamma^k)^{-1} = \nabla \phi^k \). This, along with (18), implies that

\[
\nabla \phi^k(x) = (\nabla \gamma^k)^{-1}(x) - \rho H x.
\]
Algorithm 2: DAMM-SC.

1: Initialization:
2: Each node \( i \in V \) selects \( q_i^0 \in \mathbb{R}^d \) such that \( \sum_{i \in V} q_i^0 = 0 \) (or simply sets \( q_i^0 = 0 \)).
3: Each node \( i \in V \) arbitrarily chooses \( z_i \in \mathbb{R}^d \) and sends \( z_i \) to every neighbor \( j \in N_i \).
4: Each node \( i \in V \) sets \( x_i^0 = \nabla_i \gamma(x) \) (depending on \( z_i \)
   \( \forall j \in N_i \cup \{i\} \)) and sends it to every neighbor \( j \in N_i \).
5: Each node \( i \in V \) sets \( y_i^0 = z_i - \rho \sum_{j \in N_i \cup \{i\}} p_{ij} x_j^0 \).
6: Each node \( i \in V \) sends \( y_i^0 - \nabla_i f_i(x_i^0) - q_i^0 \) to every neighbor \( j \in N_i \).
7: for \( k \geq 0 \) do
8: Each node \( i \in V \) computes \( x_i^{k+1} = \nabla_i \gamma(y_i^k - \nabla f_i(x_i^k) - q_i^k) \) (which only depends on \( y_j^k - \nabla f_j(x_j^k) - q_j^k \) \( \forall j \in N_i \cup \{i\} \)).
9: Each node \( i \in V \) sends \( x_i^{k+1} \) to every neighbor \( j \in N_i \).
10: Each node \( i \in V \) computes \( y_i^{k+1} = y_i^k - \nabla f_i(x_i^{k+1}) - q_i^k - \rho \sum_{j \in N_i \cup \{i\}} p_{ij} x_j^{k+1} \).
11: Each node \( i \in V \) computes \( q_i^{k+1} \) via (12).
12: Each node \( i \in V \) sends \( y_i^{k+1} - \nabla f_i(x_i^{k+1}) - q_i^{k+1} \) to every neighbor \( j \in N_i \).
13: end for

share \( x_i^{k+1} \) with its neighbors and the local communications in Line 6 and Line 12 of Algorithm 2 are eliminated.

Way #2: For each \( k \geq 0 \), let \( \gamma_i(x) = \frac{1}{2} x^T G_k x \), where \( G_k = (G_k)^T \in \mathbb{R}^{N_d \times N_d} \). Suppose there exist \( \bar{\gamma} \geq \gamma \geq 0 \) such that \( \gamma_{I_{N_d}} \geq G_k \geq \gamma_{I_{N_d}} \forall k \geq 0 \), and also suppose \( (G_k)^{-1} \geq \rho H \).

III. EXISTING ALGORITHMS AS SPECIALIZATIONS

This section exemplifies that AMM and its distributed realizations generalize a variety of existing distributed optimization methods originally developed in different ways.

A. Specializations of DAMM-SQ

DAMM-SQ described in Algorithm 3 generalizes several distributed first-order and second-order algorithms for solving problem (1) \( \forall i \in V \), including the exact first-order algorithm (EXTRA) [5], the inexact distributed forward-backward Bregman splitting method (ID-FBBS) [13], and the decentralized quadratically approximated ADMM (DQM) [19].

1) EXTRA: EXTRA [5] is a well-known first-order algorithm developed from a decentralized gradient descent method. From [5, (3.5)], EXTRA can be expressed as

\[
x^{k+1} = (\bar{W} \otimes I_d) x^k - \alpha \nabla f(x^k) + \sum_{i=0}^{k} (W_i - \bar{W}) \otimes I_d) x^t
\]

where \( x^0 \) is arbitrarily given, \( \alpha > 0 \), and \( W, \bar{W} \in \mathbb{R}^{N \times N} \) are two average matrices associated with \( G \). By letting \( q^k = \frac{1}{\rho} (I_N - P) \otimes I_d \) and \( P = \bar{P} \in I_N \), we can be satisfied by letting \( I_N - P = I_N - \bar{P} \) be strictly diagonally dominant with positive diagonal entries. Similar to the conclusions below Assumption 3, such \( P \) and, therefore, \( G_k \) can be locally determined by the nodes.

\[
x^{k+1} = (\bar{W} \otimes I_d) x^k - \alpha \nabla f(x^k) + \sum_{t=0}^{k} (W_t - \bar{W}) \otimes I_d) x^t
\]

where \( x^0 \) is arbitrarily given, \( \alpha > 0 \), and \( W, \bar{W} \in \mathbb{R}^{N \times N} \) are two average matrices associated with \( G \). By letting \( q^k = \frac{1}{\rho} (I_N - P) \otimes I_d \) and \( P = \bar{P} \in I_N \), we can be satisfied by letting \( I_N - P = I_N - \bar{P} \) be strictly diagonally dominant with positive diagonal entries. Similar to the conclusions below Assumption 3, such \( P \) and, therefore, \( G_k \) can be locally determined by the nodes. Moreover, since \( (I_N - P)^{-1} \geq I_N - P, (G_k)^{-1} = \rho (I_N - P)^{-1} \otimes I_d \geq H \).

\[
x^{k+1} = (\bar{W} \otimes I_d) x^k - \alpha \nabla f(x^k) + \sum_{t=0}^{k} (W_t - \bar{W}) \otimes I_d) x^t
\]

where \( x^0 \) is arbitrarily given, \( \alpha > 0 \), and \( W, \bar{W} \in \mathbb{R}^{N \times N} \) are two average matrices associated with \( G \). By letting \( q^k = \frac{1}{\rho} (I_N - P) \otimes I_d \) and \( P = \bar{P} \in I_N \), we can be satisfied by letting \( I_N - P = I_N - \bar{P} \) be strictly diagonally dominant with positive diagonal entries. Similar to the conclusions below Assumption 3, such \( P \) and, therefore, \( G_k \) can be locally determined by the nodes. Moreover, since \( (I_N - P)^{-1} \geq I_N - P, (G_k)^{-1} = \rho (I_N - P)^{-1} \otimes I_d \geq H \).

\[
x^{k+1} = (\bar{W} \otimes I_d) x^k - \alpha \nabla f(x^k) + \sum_{t=0}^{k} (W_t - \bar{W}) \otimes I_d) x^t
\]
\[ \frac{1}{\alpha} \sum_{t=0}^{k} ((\bar{W} - W) \otimes I_d) x^t, \] (23) becomes
\[ x^{k+1} = (\bar{W} \otimes I_d) x^k - \alpha \nabla f(x^k) - \alpha q^k, \] (24)
\[ q^{k+1} = q^k + \frac{1}{\alpha} ((\bar{W} - W) \otimes I_d) x^{k+1}. \] (25)

This is in the form of DAMM-SQ with \( \rho = 1/\alpha, \bar{P} = \bar{W} - W, \ P = I_N - \bar{W}, \) and \( G^k = \alpha I_N. \) As [5] assumes \( \bar{W} \geq W \) and \( \text{Null}(\bar{W} - W) = \text{span}(1), \) Assumption 3 and (11) are guaranteed. Besides, [5] assumes \( \bar{W} \succ 0, \) so that \( (G^k)^{-1} = \rho I_N \geq \rho((\bar{W} - W) \otimes I_d) = \rho H. \) It is then straightforward to see that this particular \( G^k \) satisfies all the conditions in Section II-C.

2) ID-FBBBS: ID-FBBBS [13] takes the form of (24) and (25), except that \( W = 2\bar{W} - I_N \) and \( q^0 \) can be any vector in \( S^1. \) Since [13] also assumes \( \bar{W} \succ 0, \) it follows from the analysis in Section III-A that ID-FBBBS is a particular example of DAMM-SQ, where \( \rho = 1/\alpha, \bar{P} = I_N - \bar{W}, \) and \( G^k = \alpha I_N, \) with Assumption 3 and all the conditions on \( G^k \) satisfied.

3) DQM: DQM [19] is a distributed second-order method for solving problem (1) with strongly convex, smooth, and twice continuously differentiable \( f_i \)'s and zero \( h_i \)'s. DQM takes the following form: \( x_i^{k+1} = x_i^k - (2c_i|N|_i I_d + \nabla^2 f_i(x_i^k))^{-1} (\sum_{j \in N_i} (x_i^k - x_j^k) + \nabla f_i(x_i^k) + q_i^k) \) and \( q_i^k = q_i^k + c \sum_{j \in N_i} (x_i^{k+1} - x_j^{k+1}), \) where \( x_i^k \) are arbitrarily given, \( q_i^k \) are arbitrary, and \( x_i^0 \) satisfy \( \sum_{i \in V} q_i^0 = 0, \) and \( c > 0. \) Observe that DAMM-SQ reduces to DQM if we set \( G^k = (2c_i |N|_i I_d + \nabla^2 f_i(x_i^k))^{-1}, \) \( \rho = c, \) \( p_{ij} = p_{ji}, \) and \( q_{ij} = q_{ji}. \) \( G^k \) and \( G^k \) meet the other requirements in Section II-C.

B. Specializations of DAMM

A number of distributed algorithms for composite or nonsmooth convex optimization can be cast into the form of DAMM described in Algorithm 1, including the proximal-gradient consensus method (PGC) [16], the proximal gradient exact first-order method (PG-EXTRA) [12], the distributed proximal gradient algorithm (DPGA) [11], the decentralized ADMM [4], and the distributed forward-backward Bregman splitting method (DF-BBS) [13].

1) PGC and PG-EXTRA: PGC [16] and PG-EXTRA [12] are two recently proposed distributed methods for solving problem (1), where PGC is constructed upon ADMM [32] and PG-EXTRA is an extension of EXTRA [5] to address (1) with nonzero \( h_i \)'s. According to [16], Section V-D, PGC can be described by \( q_i^0 \in S^1, q_i^k = q_i^0 + \sum_{t=1}^{k} ((A_i(W - W)) \otimes I_d) x^t \) \( \forall k \geq 1, \) and \( x^{k+1} = \arg \min_{x \in R^N} h(x) + \frac{1}{2} \|x - ((\bar{W} \otimes I_d) x^k)\|_{\Lambda_i}^2 + \langle \nabla f(x^k) + q^k, x \rangle \) \( \forall k \geq 0, \) where \( x^0 \) is arbitrarily given and the parameters are chosen as follows: Let \( \Lambda_i = \text{diag}(\beta_1, \ldots, \beta_N) \) be a positive definite diagonal matrix and \( W_i \in R_{N \times N} \) be two-row stochastic matrices such that \( \Lambda_i W_i \) and \( \Lambda_i \bar{W}_i \) are symmetric, \( W_i [i] [j] > 0 \) \( \forall j \in N_i \cup \{i\}, \) and \( W_i [i] [j] = [W_i]_{ij} = 0 \) otherwise. To cast PGC in the form of DAMM, let \( \rho = 1, P = \Lambda_i \bar{W} - W, \) and \( P = \Lambda_i(W - W). \) Then, starting from any \( q^0 \in S^1, \) the updates of PGC can be expressed as
\[ x^{k+1} = \arg \min_{x \in R^N} h(x) + \frac{1}{2} \|x - x^k\|_{A_i}^2 + \langle \nabla f(x^k) + q^k + \rho (P \otimes I_d) x^k, x \rangle, \] (26)
\[ q^{k+1} = q^k + \rho (\bar{W} \otimes I_d) x^{k+1}. \] (27)

This means that PGC is exactly in the form of DAMM with \( \psi^k(x) = \frac{\rho}{2} \|x\|^2. \) Note that \( \Lambda_i \bar{W} \geq \Lambda_i \bar{W} \) and \( \text{Null}(\Lambda_i(W - W)) = \text{span}(1). \) In addition, [16] assumes \( \Lambda_i \bar{W} \geq \Lambda_i \bar{W}, \) \( \Lambda_i \bar{W} \geq O, \) and \( \text{Null}(\Lambda_i(W - W)) = \text{span}(1). \) Consequently, Assumptions 3 and 4 hold. PG-EXTRA can also be described by (26) and (27) with \( \beta_i = \rho > 0 \) \( \forall i \in V \) and \( q^0 = \rho (\bar{W} \otimes I_d) x^0, \) i.e., is a special form of PGC. Therefore, DAMM generalizes both PGC and PG-EXTRA.

2) DPGA and Decentralized ADMM: DPGA [11] is a distributed proximal gradient method and has the following form: Given arbitrary \( x_i^0 \) and \( q_i^0 = 0, \)
\[ x_i^{k+1} = \arg \min_{x \in R^N} h_i(x) + \frac{1}{2} \|x - x_i^k\|^2 + \langle \nabla f(x_i^k) + q_i^k + \sum_{j \in N_i} (\Gamma_{ij} x_j^k), x_i \rangle, \]
\[ q_i^{k+1} = q_i^0 + \sum_{j \in N_i \setminus \{i\}} (\Gamma_{ij} x_j^{k+1}), \] \( \forall i \in V \)
where \( c_i > 0 \) \( \forall i \in V, \) \( \Gamma_{ij} = \Gamma_{ji} < 0 \) \( \forall \{i, j\} \in E, \) and \( \Gamma_{ii} = \sum_{j \in N_i \setminus \{i\}} (\Gamma_{ij} x_j^k) \forall i \in V. \) The above update equations of DPGA are equivalent to those of DAMM with \( \psi_i^k(x) = \frac{1}{2} \|x\|^2, \) \( \rho = 1, \) and \( P, \bar{P} \) such that \( \bar{p}_{ij}, p_{ij} \) is equal to \( \Gamma_{ij} \) if \( \{i, j\} \in E \) or \( i = j \) and are equal to 0 otherwise. Apparently, \( P \) and \( \bar{P} \) satisfy Assumption 3. Furthermore, due to the conditions on \( c_i \) in [11], it is guaranteed that \( \sum_{i \in V} \psi_i^k(x_i) = \frac{1}{2} \|x\|^2, \) which satisfies Assumption 4 because [13] assumes \( W \succ 0. \) Therefore, we conclude that D-FBBBS can be specialized from DAMM.

C. Specializations of AMM

Since DAMM-SQ and DAMM are subsets of AMM, the algorithms in Sections III-A and III-B are also specializations of AMM. Below, we present some other methods including the distributed ADMM [9], the distributed primal-dual algorithm [14], the distributed inexact gradient method using gradient tracking (DIGing) [25], and the exact second-order method (ESOM) [18],
which can be specialized from AMM but do not belong to DAMM, DAMM-SC, or DAMM-SQ.

1) Distributed ADMM and Distributed Primal-Dual Algorithm: In [9], a distributed ADMM is proposed to solve (1) with $f_i, \forall i \in \mathcal{V}$:

$$x^{k+1} = \arg \min_{x \in R^{N \times N}} h(x) + \langle QT w_k, x \rangle + c(Q T (\Lambda^{-1} \otimes I_d) Q x_k, x) + \frac{\alpha}{2} \|x - x^k\|_Q^2,$$

$$w^{k+1} = w_k + c(\Lambda^{-1} \otimes I_d) Q x^{k+1},$$

where $x^0 \in R^{N \times N}$ is arbitrarily given and $w^0 = 0$. In the above, $c > 0$, $\Lambda = \text{diag}(N_1, 1, \ldots, N_N + 1)$, and $Q = \Gamma \otimes I_d$ with $\Gamma \in R^{N \times N}$ satisfying $[\Gamma]_{ij} = 0 \ \forall i \in \mathcal{V} \ \forall j \notin N_i \cup \{i\}$ and $\text{Null}(\Gamma T \Lambda^{-1} \Gamma) = \text{span} \{1\}$. Additionally, $Q = \text{diag}(Q_1, \ldots, Q_N)$, $\rho = c$, and $u^k(x) = \frac{1}{2}(x - x^k)^T(Q - H)(x - x^k)$. Clearly, $H$ and $H$ satisfy the conditions in Section II-A. Also, Assumption 2 holds, since $(M + \epsilon + 2p) \text{Null}(Q K^{-1}) \geq \nabla^2 f(x^k) + c I_N + p \rho H$ [18], where $M > 0$ is the Lipschitz constant of all the $\nabla f_i$’s. Note that unlike most specializations of AMM discussed in this section, $u^k(\cdot) + \frac{1}{2} \|\cdot\|_H^2$ for ESOM-K with $K \geq 1$ is nonseparable.

2) DiGiG on Static Networks: DiGiG [25] is a distributed gradient-tracking method for solving problem (1) with $h_i, \forall i \in \mathcal{V}$ over time-varying networks. Here, we only consider DiGiG on static undirected networks. Let $\alpha > 0$ and $W \in R^{N \times N}$ satisfy $W_1 = W^T 1 = 1, [W]_{ij} = 0 \ \forall i \in \mathcal{V} \ \forall j \notin N_i \cup \{i\}$, and $\|W - \frac{1}{N} 11^T\| < 1$. It is shown in [25] that DiGiG with $W = W^T$ can be written as follows:

$$x^{k+2} = (2W \otimes I_d)x^{k+1} - (W^2 \otimes I_d)x^k - \alpha(\nabla f(x^{k+1}) - \nabla f(x^k)), \ \forall k \geq 0$$

where $x^0$ is arbitrary and $x^1 = (W \otimes I_d)x^0 - \alpha \nabla f(x^0)$. By adding the above equation from $k = 0$ to $k = K - 1$ and letting $q^0 = \frac{1}{\alpha}((W^2 - W) \otimes I_d)x^0$, the update is the same as

$$x^{K+1} = (W^2 \otimes I_d)x^K - \alpha \nabla f(x^K) - \alpha q^K,$$

$$q^{K+1} = q^K + \frac{1}{\alpha}((I_N - W^2) \otimes I_d)x^{K+1}.$$

Such an algorithmic form of DiGiG is identical to AMM (9)–(11) with the above given $q^0 \in S^2$, $p = 1/\alpha$, $H = (I_N - W^2) \otimes I_d$, $H = (I_N - W)^2 \otimes I_d$, and $u^k(x) = \langle \nabla f(x^k), x \rangle + \frac{p}{2}(W \otimes I_d)(x - x^k)^2$. It can be verified that $u^k \forall k \geq 0$ and $H, H$ satisfy all the conditions in Section II-A.

3) ESOM: ESOM [18] is a class of distributed second-order algorithms that address problem (1) with $f_i, \forall i \in \mathcal{V}$ being strongly convex, smooth, and twice continuously differentiable and with $h_i, \forall i \in \mathcal{V}$. It is developed by incorporating a proximal technique and certain second-order approximations into the method of multipliers. To describe ESOM, let $\alpha > 0, \epsilon > 0, \tau > 0, W \in R^{N \times N}$ be an average matrix associated with $G$ such that $[W]_{ij} \geq 0 \ \forall i, j \in \mathcal{V}$. In addition, define $D := \text{diag}([W]_{11}, \ldots, [W]_{NN}), B := \alpha(I_{N\Delta} + (W - 2W_d) \otimes I_d)$, $D_k := \nabla^2 f(x^k) + c I_N + 2\alpha(I_{N\Delta} - W_d \otimes I_d)$, and $Q_k(I) := (D_k)^{-\frac{1}{2}}\sum_{\tau \leq 0}((D_k)^{-\frac{1}{2}}B(D_k)^{-\frac{1}{2}})^{\tau}(D_k)^{-\frac{1}{2}}, K \geq 0$. With the above notations, each ESOM-K algorithm can be described by

$$x^{k+1} = x^k - Q_k(I)(\nabla f(x^k) + q^k + \alpha(I_{N\Delta} - W \otimes I_d)x_k), (28)$$

$$q^{k+1} = q^k + \alpha(I_{N\Delta} - W \otimes I_d)x^{k+1}, (29)$$

where $x^0$ is any vector in $R^{N \Delta}$ and $q^0 = 0$ which satisfies the initialization (11) of AMM. Note that the primal and dual updates of AMM, i.e., (9) and (10), reduce to (28) and (29) when $H = H = I_{N\Delta} - W \otimes I_d, \rho = \alpha$, and $u^k(x) = \frac{1}{2}(x - x^k)^T(Q_k(I))^{-1} - \rho H)(x - x^k)$ such that $Q_k(I) = (Q_1, \ldots, Q_N)$ and $A := I_{N\Delta} - W \otimes I_d$ is nonseparable.

D. Connections to Existing Unifying Methods

The proximal unified decentralized algorithm (PUDA) [22] and the general primal-dual proximal algorithm (ABC) [21] are two recently proposed distributed methods for convex composite optimization, which unify a number of existing methods including a subset of the aforementioned AMM’s specializations. Nevertheless, unlike AMM that can be specialized to both first-order and second-order methods, PUDA and ABC are first-order algorithms. Moreover, AMM allows $h_i, \forall i \in \mathcal{V}$ to be nonidentical, i.e., each node only needs to know a local portion of the global nonsmooth objective $h$. In contrast, PUDA and ABC are restricted to the case of identical $h_i$’s, i.e., each node knows the entire $h$, and it is not straightforward to extend their analyses to the more general case of nonidentical $h_i$’s.

Although none of AMM, PUDA, and ABC can include the others as special cases, they are implicitly connected through the following algorithm: Let $q^0 = 0_{N \Delta}$. For any $k \geq 0$, let

$$z^{k+1} = \arg \min_{z \in R^{N \Delta}} u^k(z) + \frac{\rho}{2} \|z\|_H^2 + \|q^k\|_T^2, (30)$$

$$x^{k+1} = \arg \min_{x \in R^{N \Delta}} h(x) + \frac{\rho}{2} \|x - z^{k+1}\|_H^2, (31)$$

$$q^{k+1} = q^k + \rho H z^{k+1}. (32)$$

Different from AMM whose primal update (9) involves both the surrogate function and the nonsmooth objective, the above algorithm tackles these two parts separately and thus has two sequential primal minimization operations. Indeed, when $h = 0$, (30)–(32) are identical to (9) and (10).

With particular $u^k, H, H, (30)$–(32) and $q^0 = 0_{N \Delta}$ become equivalent to PUDA and ABC under some mild parameter conditions. Specifically, PUDA corresponds to $u^k(z) = \frac{1}{2} \|z - x^k\|_C^2 + (\nabla f(x^k), z)$, $\rho > 0, H = A_1^{-1} - I_{N\Delta} + C_1$, and $H = B_2^{-1} A_1^{-1}$, while ABC corresponds to $u^k(z) = \frac{1}{2} \|z - x^k\|_C^2 + (\nabla f(x^k), z), \rho > 0, H = B_2^{-1}(I_{N\Delta} - A_2), \text{and } H = B_2^{-1} C_2$. Here, $A_1, B_1, C_1,$
A. Under General Convexity

In this subsection, we provide the convergence rates of AMM in solving the general convex problem (1).

Let $x_k^* = \frac{1}{k} \sum_{t=1}^{k} x^t \forall k \geq 1$ be the running average of $x_t$ from $t = 1$ to $t = k$. Below, we derive sublinear convergence rates for 1) the consensus error $\|H \hat{x}_k^*\|$, which quantifies the indefeasibility of $x_k^*$ in solving the equivalent problem (3), and 2) the objective error $|f(\hat{x}_k^*) + h(\hat{x}_k^*) - f(x^*) - h(x^*)|$, which is a direct measure of optimality. Throughout Section IV-A, we tentatively consider the following type of surrogate function that fulfills Assumption 2:

$$u_k(x) = \frac{1}{2} \|x - x^k\|^2_A + (\nabla f(x^k), x), \forall k \geq 0$$

where $A \in \mathbb{R}^{N_d \times N_d}$ satisfies $A = A^T$, $A \succeq 0$, and $A + \rho H \succ 0$. Such a choice of $u_k$ results in $A^k = A \forall k \geq 0$.

Note that AMM endowed with (35) still generalizes most existing algorithms discussed in Section III, including EXTRA [5], ID-FBBS [13], PGC [16], PG-EXTRA [12], DPGA [11], the decentralized ADMM in [4], D-FBBS [13], the distributed ADMM in [9], the distributed primal-dual algorithm in [14], and DI Gin [27]. Although DQM [19] and ESOM [18] are specialized from AMM with different $u_k$'s other than (35), they both require problem (1) to be strongly convex and smooth—in such a case, we will provide convergence rates for the general form of AMM in Section IV-B.

Now we bound $\|H \hat{x}_k^*\|$ and $|f(\hat{x}_k^*) + h(\hat{x}_k^*) - f(x^*) - h(x^*)|$. This plays a key role in acquiring the rates at which these errors vanish.

**Lemma 2:** Suppose Assumption 1, Assumption 6, and (35) hold. For any $k \geq 1$,

$$\|H \hat{x}_k^*\| \leq \frac{1}{\rho k} \|v^k - v^0\|, \quad (36)$$

$$f(\hat{x}_k^*) + h(\hat{x}_k^*) - f(x^*) - h(x^*) \leq \frac{1}{k^2} \frac{\|v^0\|^2}{2\rho} + \frac{\|x^0 - x^*\|^2_2}{2A} + \sum_{t=0}^{k-1} \|x^{t+1} - x^t\|^2_{\Lambda_{M-A}}, \quad (37)$$

$$f(\hat{x}_k^*) + h(\hat{x}_k^*) - f(x^*) - h(x^*) \geq \frac{\|v^*\| - \|v^k - v^0\|}{\rho k}. \quad (38)$$

**Proof:** See Appendix B.

Observe from Lemma 2 that as long as $\|v^k - v^0\|$ and $\sum_{t=0}^{k-1} \|x^{t+1} - x^t\|^2_{\Lambda_{M-A}}$ are bounded, $\|H \hat{x}_k^*\|$ and $|f(\hat{x}_k^*) + h(\hat{x}_k^*) - f(x^*) - h(x^*)|$ are guaranteed to converge to 0. The following lemma asserts that this is true with the help of Lemma 1, in which $x_k^* := ((x_k^*)_T, (v_k^*)^T) \forall k \geq 0$, $z^* := ((x^*)_T, (v^*)^T)_T$, and $G := \text{diag}(A, I_{N_d/\rho})$.

**Lemma 3:** Suppose all the conditions in Lemma 2 hold. Also suppose that $A \succcurlyeq \Lambda_{M/2}$. For any $k \geq 1$,

$$\|v^k - v^0\| \leq \|v^0 - v^*\| + \sqrt{7} \|z^0 - z^*\|_G, \quad (39)$$

$$\sum_{t=0}^{k-1} \|x^{t+1} - x^t\|^2_{\Lambda_{M-A}} \leq \frac{\|z^0 - z^*\|^2_G}{1 - 2\varrho}, \quad (40)$$

where $\varrho := \min\{\sigma \mid \sigma \Lambda \succeq \Lambda_{M/2} \in (0, 1)$. 

In this section, we analyze the convergence performance of AMM described by (4) and (5) [equivalent to (9)–(11)].

For the purpose of analysis, we add $H \succeq \hat{H}$ to the conditions on $H, \hat{H}$ in Section II-A, leading to Assumption 6.

**Assumption 6:** The matrices $H$ and $\hat{H}$ are symmetric and $\text{Null}(H) = \text{Null}(\hat{H}) = S$, where $S$ is defined in (2). In addition, $H \succeq \hat{H} \succeq 0$.

In DAMM, DAMM-SC, and DAMM-SQ, we adopt $H = P \otimes I_d$ and $\hat{H} = \hat{P} \otimes I_d$, where $P$ and $\hat{P}$ comply with Assumption 3. Thus, as long as we further impose $\hat{P} \succeq P$, Assumption 6 holds. Moreover, all the existing specializations of DAMM, AMM, and DAMM-SQ in Section III, except DI Gin [25], also need to satisfy Assumption 6. DI Gin is not necessarily required to meet $H \succeq \hat{H}$; yet, it can easily satisfy $H \succeq \hat{H}$ by letting its average matrix be symmetric positive semidefinite.

In what follows, we let $(x^k, v^k)$ be generated by (4) and (5), and let $(x^*, v^*)$ be a primal-dual optimal solution pair of problem (3), which satisfies (7). Also, let $\Lambda := \text{diag}(M_1, \ldots, M_N) \otimes I_d$, where $M_i > 0$ is the Lipschitz constant of $\nabla f_i$. In addition, define $A_k := \int_0^1 \nabla^2 u_k((1-s)x^k + sx^{k+1})ds$, \forall k \geq 0.

Note that every $A_k$ exists and is symmetric due to Assumption 2. Moreover, since $\nabla^2 u_k(x^k) = \nabla f(x^k),

$$\nabla u_k(x^{k+1}) = \nabla u_k(x^k) + \int_0^1 \nabla^2 u_k((1-s)x^k + sx^{k+1})(x^{k+1} - x^k)ds = \nabla f(x^k) + A_k(x^{k+1} - x^k). \quad (33)$$

Then, we introduce the following auxiliary lemma.

**Lemma 1:** Suppose Assumption 1, Assumption 2, and Assumption 6 hold. For any $\eta \in [0, 1)$ and $k \geq 0$,

$$\frac{1}{\rho} (v_k - v^{k+1}, v^{k+1} - v^*) \succeq \eta (x_k - x^*, \nabla f(x^k) - \nabla f(x^*)) + (x^{k+1} - x^*, A_k(x^{k+1} - x^k)) - \frac{\|x^{k+1} - x^k\|_A^2}{4(1 - \eta)}. \quad (34)$$

**Proof:** See Appendix A.
Proof: See Appendix C.

The following theorem results from Lemmas 2 and 3, which provides $O(1/k)$ rates for both the consensus error and the objective error at the running average $\bar{x}^k$.

**Theorem 1:** Suppose all the conditions in Lemma 3 hold. For any $k \geq 1$,

\[
\begin{align*}
\|\bar{H}^k \bar{x}^k\| &\leq \left\|\bar{v}^0 - \bar{v}^*\right\| + \sqrt{\rho} \left\|z^0 - z^*\right\|_G, \\
 f(\bar{x}^k) + h(\bar{x}^k) - f(\bar{x}^*) - h(\bar{x}^*) &\leq \frac{1}{2k} \left( \left\|\bar{v}^0\right\|^2 + \left\|z^0 - z^*\right\|^2 + \frac{\left\|z^0 - z^*\right\|^2}{1 - \rho} \right), \\
 f(\bar{x}^k) + h(\bar{x}^k) - f(\bar{x}^*) - h(\bar{x}^*) &\geq -\frac{\left\|\bar{v}^0 - \bar{v}^*\right\| + \sqrt{\rho} \left\|z^0 - z^*\right\|_G}{\rho k},
\end{align*}
\]

Proof: Substitute (39) and (40) into (36)-(38).

Among the existing distributed algorithms that are able to solve nonsmooth convex optimization problems with non-smooth convex objective functions (e.g., [6], [7], [9]–[13], [16], [17], [21], [28]–[31]), the best convergence rates are of $O(1/k)$, achieved by [7], [9], [11]–[13], [16], [21], and [30]. Indeed, AMM with all the conditions in Theorem 1 generalizes the algorithms in [9], [11]–[13], and [16] as well as their parameter conditions. Like Theorem 1, [9], [11], [16], and [21] achieve $O(1/k)$ rates for both the objective error and the consensus error, whereas [7], [12], [13], and [30] reach $O(1/k)$ rates in terms of optimality residuals that implicitly reflect deviations from an optimality condition, and they only guarantee $O(1/\sqrt{k})$ rates for the consensus error. Also note that [21] and [30] only address problem (1) with identical $h_i$'s.

Theorem 1 provides new convergence results for some existing algorithms generalized by AMM. In particular, the $O(1/k)$ rate in terms of the objective error is new to PG-EXTRA [12] and D-FBBS [13] for nonsmooth problems as well as EXTRA [5] and D-FBBS [13] for smooth problems, which only had $O(1/k)$ rates with respect to optimality residuals before. Moreover, Theorem 1 improves their $O(1/\sqrt{k})$ rates in reaching consensus to $O(1/k)$. Furthermore, Theorem 1 extends the $O(1/k)$ rate of the distributed primal-dual algorithm in [14] for constrained smooth convex optimization to nonsmooth convex problems and allows PGC [16] to establish its $O(1/k)$ rate without the assumption that each $h_i$ has a compact domain.

Finally, we illustrate how the sublinear rates in Theorem 1 are influenced by the graph topology. For simplicity, set $v^0 = 0$ and $\bar{H} \preceq I_{Nd}$. Also, let $g^* \in \partial h(x^*)$ be such that $\nabla f(x^*) + g^* \in S^1$, which exists because of the optimality condition $-\sum_{i \in V} \nabla f_i(x^*) \in \partial \sum_{i \in V} h_i(x^*)$, and let $v^* = -(\bar{H}^k)^{-1}(\nabla f(\bar{x}) + g^*)$. It follows from (7) that $v^*$ is an optimal solution to the Lagrange dual of (3). Then, $\left\|v^0 - v^*\right\| = \left\|v^*\right\| \leq \frac{1}{\sqrt{\lambda_{\bar{H}}}} \left\|\nabla f(x^*) + g^*\right\|$, where $\lambda_{\bar{H}} \in (0, 1]$ is the smallest nonzero eigenvalue of $\bar{H}$. Note that $\left\|z^0 - z^*\right\|_A = \left\|\left\|x^0 - x^*\right\|_A + \frac{1}{\rho} \left\|v^0 - v^*\right\|^2\right\| \leq \left\|x^0 - x^*\right\|_A + \frac{1}{\sqrt{\rho}} \left\|v^0 - v^*\right\|$. Thus, by letting $\rho = \frac{1}{\sqrt{\lambda_{\bar{H}}}}$,

we have $\|f(\bar{x}) + h(\bar{x}) - f(x^*) - h(x^*)\| \leq O\left(\frac{1}{k^{1/2}}\right)$. In addition, $\|P_{S^c} f(x^*)\| \leq \frac{1}{\sqrt{\lambda_{\bar{H}}}} \left\|\nabla f(\bar{x}) + g^*\right\| \leq O\left(\frac{1}{k^{1/2}}\right)$. Here, $\|P_{S^c} f(x^*)\|$ plays a similar role as $\|\bar{H}^k \bar{x}^k\|$ in quantifying the consensus error. Therefore, denser network connectivity, which is often indicated by larger $\lambda_{\bar{H}}$, can yield faster convergence of both the objective error and the consensus error (measured by $\|P_{S^c} f(x^*)\|$).

Compared to the existing specializations of AMM that also have $O(1/k)$ rates in solving (1), the above $O(1/(k^{1/2}/\sqrt{\lambda_{\bar{H}}}))$ rate has better dependence on $\lambda_{\bar{H}} \in (0, 1)$ than the $O(1/(k\lambda_{\bar{H}}^{1/2}))$ rate of DPGA [11] and the $O(1/(k\lambda_{\bar{H}}^{1/2}))$ rate of the distributed ADMM [9]. The remaining specializations discussed in Section III have not been provided with similar dependence of their convergence rates on the graph topology.

**B. Under Local Restricted Strong Convexity**

In this subsection, we additionally impose a problem assumption and acquire the convergence rates of AMM. Henceforth, we no longer restrict $u^k$ to the form of (35).

**Assumption 7:** The function $\sum_{i \in V} f_i(x)$ is locally restricted strongly convex with respect to $x^*$, where $x^*$ is an optimum of problem (1), i.e., for any convex and compact set $C \subseteq \mathbb{R}^d$ containing $x^*$, $\sum_{i \in V} f_i(x)$ is restricted strongly convex with respect to $x^*$ on $C$ with convexity parameter $m_C > 0$.

Restricted strong convexity has been fairly studied in the literature. For example, [5] and [23] consider its global version and [20] considers the local version as Assumption 7. Under Assumption 7, problem (1) has a unique optimum $x^*$, and problem (3) has a unique optimum $x^* = ((x^*)^T, \ldots, (x^*)^T)^T \in S$ [20]. Furthermore, Assumption 7 is less restricted than the standard global strong convexity condition. For instance, the objective function of logistic regression [36] is not globally strongly convex but meets Assumption 7.

**Proposition 1:** Suppose Assumption 1 holds. If there exists $\epsilon > 0$ such that each $f_i$ is twice continuously differentiable on the ball $B(x^*, \epsilon) := \{x : \|x - x^*\| \leq \epsilon\}$ and $\sum_{i \in V} \nabla^2 f_i(x^*)$ is positive definite, then Assumption 7 holds.

**Proof:** See Appendix D.

Proposition 1 provides a sufficient condition for Assumption 7. When each $f_i$ is twice continuously differentiable, this sufficient condition is much weaker than global strong convexity which requires $\sum_{i \in V} \nabla^2 f_i(x^*) > 0 \forall x^* \in \mathbb{R}^d$.

Note that Assumption 7 is unable to assert any kind of strong convexity of $f(x) = \sum_{i \in V} f_i(x_i)$; yet, it guarantees the following property on $f(x) + \frac{\epsilon}{2} \|x\|^2$.

**Proposition 2:** [20, Lemma 1] Under Assumptions 6 and 7, $f(x) + \frac{\epsilon}{2} \|x\|^2$ is locally restricted strongly convex with respect to $x^*$.

Subsequently, we construct a convex and compact set $C \subseteq \mathbb{R}^{Nd}$ containing $x^*$, so that the restricted strong convexity in Proposition 2 takes into effect on $C$, leading to a parameter condition that guarantees $x^* \in C \forall k \geq 0$. To introduce $C$, note from Assumption 2 that there exist symmetric positive semidefinite matrices $A_i, A \in \mathbb{R}^{Nd \times Nd}$ such that for any $x \in \mathbb{R}^{Nd}$ and $k \geq$
0, $A_t \preceq \nabla^2 u^k(x) \preceq A_u$. Let $\Delta := \|A_u - A_t\| \geq 0$. Moreover, define $A^e = (A_t + A_u)/2$ and $\tilde{G} = \text{diag}(A^e, I_{N_d}/\rho)$, which are also symmetric positive semidefinite. Then, let $C := \{x : \|x - \tilde{x}\|_{\tilde{G}}^2 + \rho\|x^k\|_{\tilde{H}}^2 \leq \|z^k - z^*\|_{\tilde{G}}^2 + \rho\|x^k\|_{\tilde{H}}^2\}$, where $z^k$ and $z^*$ are defined above Lemma 3. From Assumptions 2(b) and 6, $A_u + \rho \bar{H} \succeq \mathbf{O}$, so that $C$ is convex and compact. Thus, from Proposition 2, $f(x) + \frac{\rho}{4}\|x\|_{\bar{H}}^2$ is restricted strongly convex with respect to $x^*$ on $C$ with convexity parameter $m_p \in (0, \infty)$. Lemma 4 is another consequence of Lemma 1, showing that $x^k$ stays identically in $C$.

**Lemma 4:** Under Assumptions 1, 2, 6, and 7, $x^k \in C \forall k \geq 0$ provided that

$$A_u \geq \left(\frac{\Delta^2}{8\eta m_p + \Delta}\right) I_{N_d} + \frac{\Lambda_M}{2(1-\eta)} \text{ for some } \eta \in (0, 1)$$

(41)

where $\Lambda_M$ is defined above Lemma 1.

**Proof:** See Appendix E.

When $\Delta = 0$ (which means $\nabla^2 u^k$ is constant) or $\sum_{i \in V} J_i(x)$ is globally restricted strongly convex (which means $m_p\zeta$ can be independent of $C$), it is straightforward to find $u^k$ so that (41) holds. Otherwise, $m_p$ depends on $C$ and thus on $A_u$, so that both sides of (41) involve $A_u$. With that being said, (41) can always be satisfied by proper $u^k$s. To see this, arbitrarily pick $\eta \in (0, 1)$ and $\tilde{\lambda}_a, \tilde{\lambda}_b > 0$ such that $\tilde{\lambda}_a > \tilde{\lambda}_b > \frac{M}{\rho |x|^2 \eta}$, where $M := \sum_{i \in V} M_i$. If we choose $u^k$ such that the corresponding $A_u$ satisfies $\tilde{\lambda}_a I_{N_d} \preceq A_u \preceq \tilde{\lambda}_b I_{N_d}$, then $C$ is a subset of $C' := \{x : \frac{\eta}{\tilde{\lambda}_a} \|x - x^*\|^2 \leq \frac{\tilde{\lambda}_b}{\tilde{\lambda}_a} \|x\|^2 + \frac{\rho}{4}\|x - x^*\|^2 + \rho\|x\|^2 \leq \frac{\tilde{\lambda}_b}{\tilde{\lambda}_a} \|x\|^2 + \frac{\rho}{4}\|x\|^2\}$. Let $m'_p \in (0, m_p)$ be any convexity parameter of $f(x) + \frac{\rho\|x\|^2}{4}$ on $C'$. Clearly, $m'_p \in (0, m_p)$ and is independent of $A_u$. Then, the following suffices to satisfy (41):

$$\left(\frac{\Delta^2}{8\eta m_p + \Delta} + \frac{\Lambda_M}{2(1-\eta)}\right) I_{N_d} \preceq A_u \preceq \tilde{\lambda}_a I_{N_d}.$$ 

(42)

The lower and upper bounds on $A_u$ in (42) do not need to depend on $A_u$. Also, (42) is well-posed, since $\frac{\Delta^2}{8\eta m_p + \Delta} + \frac{\Lambda_M}{2(1-\eta)} \preceq \tilde{\lambda}_a$ holds for sufficiently small $\Delta > 0$. Therefore, $u^k \forall k \geq 0$ with such $\Delta$ and with $A_u$ satisfying (42) meet (41).

Next, we present the convergence rates of $\text{AMM}$ under local restricted strong convexity. We first consider the smooth case of (1) and provide a linear convergence rate for $\|z^k - z^*\|_{\tilde{G}}^2 + \rho\|x^k\|_{\tilde{H}}^2$, which quantifies the distance to primal optimality, dual optimality, and primal feasibility of (3). In Theorem 2, we force $v^* \in C$ to satisfy not only (7) but $v^* - v^0 \in S^+_{1,2}$ as well. Such $v^*$ is a particular value in the Lagrange dual of (3), and can be chosen as $v^* = \bar{v}^\top \left(\left(\frac{1}{T} \sum_{i \in V} (v_i - \bar{v}_i)^2\right), \ldots, \left(\frac{1}{T} \sum_{i \in V} (v_i - \bar{v}_i)^2\right)\right)$, where $\bar{v}$ is any Lagrange dual optimal of (3) and $v^0 = \left(\frac{1}{T} \sum_{i \in V} (v_i - \bar{v}_i)^2\right)$ is the initial dual iterate of $\text{AMM}$.

**Theorem 2:** Suppose all the conditions in Lemma 4 hold. Also suppose $h_i(x) = 0 \forall i \in V$. Then, there is $\tilde{\delta} \in (0, 1)$ such that

$$\|z^{k+1} - z^*\|_{\tilde{G}}^2 + \rho\|x^{k+1}\|_{\tilde{H}}^2 \leq (1-\tilde{\delta})(\|z^k - z^*\|_{\tilde{G}}^2 + \rho\|x^k\|_{\tilde{H}}^2), \quad \forall k \geq 0.$$

(43)

Specifically, $\tilde{\delta} = \max\{\delta, B_3(\delta) \geq O, \forall i \in \{1, 2, 3\}\}$ with

$$B_1(\delta) = (2\rho \eta m_p - \delta) I_{N_d} - \delta A_u - \delta(1 + \theta_1)(1 + \theta_2)\frac{\Lambda_M}{\rho \lambda \bar{H}},$$

$$B_2(\delta) = (\rho - \delta - \eta) \tilde{H},$$

$$B_3(\delta) = (1-\sigma)A_u - 2\delta(1 + \theta_1)(1 + \theta_2)\left(\frac{\Delta^2}{8\eta m_p + \Delta} + \frac{\rho \lambda \bar{H}}{\rho \lambda \bar{H}}\right)$$

where $\theta_1, \theta_2$ are arbitrary positive scalars, $\lambda$ is any upper bound on $|\bar{H}|$, $\eta \in (0, 1)$ is given in (41), and $\sigma > 0$ and $\sigma \in (0, 1)$ are such that $\beta \lambda < 2\rho m_p, \sigma A_u \geq \left(\frac{1}{4\delta} + 1\right)\Delta I_{N_d} + \frac{\Lambda_M}{2(1-\eta)}$.

**Proof:** See Appendix F.

The linear convergence in Theorem 2 is obtained under local restricted strong convexity, which is weaker than the standard global strong convexity condition assumed in [4], [6], [7], [9], [18], [19], [21], [22], and [25]–[27] to derive linear convergence rates. The gradient-tracking-based decentralized stochastic gradient descent algorithm (GT-DSGD) [37] achieves linear convergence for smooth nonconvex problems under the Polyak–Lojasiewicz (PL) condition that generalizes global restricted strong convexity but cannot generalize local restricted strong convexity. Moreover, Theorem 2 provides new convergence results to a number of existing algorithms that can be specialized from $\text{AMM}$. Specifically, it establishes linear convergence for D-FBBBS [13], ID-FBBBS [13], and DPGA [11], which has never been discussed before. In addition, it relaxes the global restricted/global strong convexity condition required by EXTRA [5], DIging [25], ESOM [18], DQM [19], the distributed ADMMs [4], [9], ABC [21], and PUDA [22] and eliminates the Lipschitz continuity condition on $\nabla^2 u$.

The above discussions suggest that smaller condition number $\kappa_f := M/m$ or denser network connectivity may lead to faster convergence of $\text{AMM}$ in solving strongly convex and smooth problems. Indeed, for some particular forms of $\text{AMM}$, such dependence can be more explicit and is comparable to some prior results. For example, let $H = \tilde{H} = (I_N - W) \otimes I_d/4$ with an average matrix $W$ associated with
$G$, $\rho = M \sqrt{2/\lambda_2}$, and $u^k$ be such that $\Delta = 0$ and $A_\delta = \rho \frac{3M + W}{\rho^2 M} \leq L_2|k|$. Thus, $\rho I_{Nd} \geq A_\delta \geq \frac{\rho}{2} I_{Nd}$, $\|H\| \leq \tilde{\lambda} = \frac{1}{2}$, and $\lambda_\delta = \frac{1}{2} - 2\beta(W) \in (0, \frac{1}{2})$, where $\lambda_2(W)$ is the second largest eigenvalue of $W$. Accordingly, $\rho \geq 2M$ and $A_\delta \geq M I_{Nd}$. We may pick $\eta = \frac{1}{2}$ and $\sigma = \frac{3}{4}$ so that (44) holds. Then, by choosing $\theta_1 = \theta_2 = 1$ in Theorem 2, we can show that
$$\tilde{\delta} = O(\min\{\frac{1}{\sqrt{\lambda_2(W)}}, 1 - \lambda_2(W)\}).$$

Most existing linearly convergent specializations of AMM have not revealed such explicit relations as (45), except the following. The decentralized ADMM [4] has the same result as (45), and DiGing [25] gives $\tilde{\delta} = O(\frac{(1 - \lambda_2(W))^2}{\sigma})$ when $W = W^T \geq 0$, which is worse than (45). The distributed ADMM [9] can reach a better $\tilde{\delta} = O\left(\frac{(1 - \lambda_2(W))^2}{\sigma}\right)$. This is probably because (45) is directly from our analysis for more general problems and algorithmic forms. Besides, when the PL condition in [37] reduces to global restricted strong convexity, the linear rate of GT-DSGD [37] with $W = W^T \geq 0$ gives $\tilde{\delta} = O\left(\frac{\min\{1, \lambda_2(W)\}}{\sigma}\right)$ (49) of $\lambda_2(W)$ is close to 1 for large-scale networks), which is essentially outperformed by (45).

Finally, we remove the condition of $h_i = 1 \forall i \in V$ in Theorem 2 and derive the following result.

**Theorem 3:** Suppose all the conditions in Lemma 4 hold. For any $k \geq 1$,
$$\|H^{1/2} x_k\| \leq \frac{d_0 + \|y^0 - y^*\|}{\rho k},$$

$$f(x^k) + h(x^k) - f(x^*) - h(x^k) \leq \frac{\Delta}{\rho},$$

$$f(x^k) + h(x^k) - f(x^*) - h(x^k) \geq -\frac{\|y^0 - y^*\|^2}{\rho k}.$$  

where $d_0 = \sqrt{\rho\|x^0 - x^*\|^2} \left| \frac{\rho^2}{\rho} \left| \frac{\rho^2}{\rho} \right| \right|_{\eta \in (0, 1)}$ is given in (41), and $\beta > 0$ and $\sigma \in (0, 1)$ are such that (44) holds.

**Proof:** See Appendix G.

The convergence rates in Theorem 3 are of the same order as those in Theorem 1. Theorem 1 considers a more general optimization problem, while Theorem 3 allows for more general $u^k$s. In the literature, [8], [10], [15], and [24] also deal with nonsmooth, strongly convex problems. Even under global strong convexity, [15] and [24] provide convergence rates slower than $O(1/k)$, and [8] derives an $O(1/k)$ rate of convergence only to dual optimality. Although [10] attains linear convergence, it considers a more restricted problem, which assumes $f_i$ to be globally restricted strongly convex and the subgradients of $h_i$ to be uniformly bounded.

**Remark 1:** Compared to the existing algorithms in Section III that can be viewed as specializations of AMM, AMM is able to achieve convergence rates of the same or even better order under identical or weaker problem conditions. Moreover, although the theoretical results in Section IV require additional conditions on the parameters of AMM besides those in Section II, these parameter conditions are not restrictive. Indeed, when AMM reduces to the algorithms in Section III, its parameter conditions in Theorem 1 still generalize those in [5], [9], [11]–[13], [14], and [16], and partially overlap those in [25] for non-strongly convex problems. The parameter conditions in Theorem 2 are more general than those in [5] and [19], different from but intersecting with those in [18] and [25], and, admittedly, relatively limited compared to those in [4] and [9] for strongly convex problems, yet on the premise that $u^k$ is specialized to the corresponding particular forms specified in Section III.

**V. NUMERICAL EXAMPLES**

This section demonstrates the competent practical performance of DAMM via numerical examples.

**A. Nonidetntical Local Nonsmooth Objectives**

Consider the following constrained $l_1$-regularized problem:
$$\min_{x \in \cap_i V_i} \sum_{i \in \mathcal{V}} \left( \frac{1}{2} \|B_i x - b_i\|^2 + \frac{1}{N} \|x\|_1 \right)$$

where each $B_i \in \mathbb{R}^{m \times d}$, $b_i \in \mathbb{R}^m$, and $X_i = \{x \mid \|x - a_i\| \leq \|a_i\| + 1\}$ with $a_i \in \mathbb{R}^d$. Note that $\cap_i V_i$ contains 0 and is nonempty. We set $f_i(x) = \frac{1}{2} \|B_i x - b_i\|^2$ and $h_i(x) = \frac{1}{N} \|x\|_1 + I_{X_i}(x) \forall i \in \mathcal{V}$. We choose $N = 20$, $d = 5$, and $m = 3$. Besides, we randomly generate each $B_i$, $h_i$, and $a_i$. The graph $G$ is also randomly generated and contains 26 links.

The simulation involves PG-EXTRA [12], D-FBBS [13], DPGA [11], and the distributed ADMM [9], which are guaranteed to solve (49) with distinct $h_i$'s. From Section III, they are specializations of AMM, and the first three algorithms can also be specialized from DAMM. In addition to these existing methods, we run a particular DAMM with $\psi_i^k(\cdot) = \psi_i(\cdot) := \frac{1}{2 \sqrt{\beta_i}} \|H_i T B_i + \epsilon L_i\|_1 > 0$, which depends on the problem data. Note that this is a new algorithm for solving (49). All these algorithms have similar computational complexities per iteration, while each iteration of the distributed ADMM requires twice the communication cost of the others. The algorithmic forms of PG-EXTRA, D-FBBS, DPGA, and the distributed ADMM are given in Section III. For DPGA, we let $c_i = c \forall i \in \mathcal{V}$ for some $c > 0$ and $\Gamma_{ij} = \frac{1}{2} [M G_{ij}] \forall i \in \mathcal{V} \forall j \in \mathcal{N}_i \setminus \{i\}$, where $M_G$ is the Metropolis weight matrix defined below Assumption 3. We also set the weight matrix $\Gamma = \frac{1}{2} G$ in the distributed ADMM. We assign $P = \frac{1}{2} G$ to the above new DAMM and to PG-EXTRA and D-FBBS when cast in the form of DAMM. The remaining parameters are all fine-tuned in their theoretical ranges to achieve the best possible performance.

Fig. 1 plots the optimality error $\|f(\cdot) + h(\cdot) - f(x^\ast) - h(x^\ast)\| + \|H^{1/2}(\cdot)\|$ which includes both the objective error and the consensus error] at the running average $x^k$ and the iterate $x^k$, respectively. For each of the aforementioned algorithms, the running average produces a smoother curve, while the iterate converges faster. For the new DAMM with $\psi_i^k(\cdot) = \psi_i(\cdot)$ outperforms the other four methods, suggesting that AMM not only unifies a diversity of existing methods but also creates
novel distributed algorithms that achieve superior performance in solving various convex composite optimization problems.

B. Methods Using Surrogate Functions

This subsection compares the convergence performance of the particular DAMM with \( \psi_{i}^{\ast} (\cdot) = \omega_{i} (\cdot) \) in Section V-A with that of the in-network successive convex approximation method (NEXT) [28] and the successive convex approximation algorithm (SONATA) [30], which also utilize surrogate functions. However, NEXT and SONATA are only guaranteed to address problem (1) with identical \( h_{i} \)’s. Thus, we change \( X_{i} \) \( \forall i \in \mathcal{V} \) in (49) to the unit ball \( \{ x \mid \| x \| \leq 1 \} \). The remaining settings comply with Section V-A.

At each iteration, the computational complexities of all the three algorithms are roughly the same; yet, the communication costs of NEXT and SONATA double that of the particular DAMM. For NEXT and SONATA, we follow the simulations in [28] and [30] to approximate \( f_{i} (x_{i}) \) by \( f_{i} (x_{i}) + \frac{\rho}{2} \| x_{i} - x_{k} \|^{2} \) for some \( \tau > 0 \) at each iteration \( k \), and choose the average matrix as \( I_{N} - M_{G} \), where \( M_{G} \) is the Metropolis weight matrix. The diminishing step-size of NEXT is set as \( c / k \) with \( c > 0 \), and SONATA adopts a fixed step-size. Again, we fine-tune all the algorithm parameters within their theoretical ranges.

Fig. 2 displays the optimality error \( f (x^{k}) + h(x^{k}) - f(x^{\ast}) - h(x^{\ast}) + \| H^{2} x^{k} \| \) generated by the aforementioned algorithms. Observe that the particular DAMM converges much faster than the other two methods. The main reason for the slow convergence of SONATA in solving this numerical example is that its theoretical step-size is very small.

VI. CONCLUSION

We have introduced the unifying AMM that emulates the method of multipliers via a surrogate function. Proper designs of the surrogate function lead to a wealth of distributed algorithms for solving convex composite optimization over undirected graphs. Sublinear and linear convergence rates for AMM are established under various mild conditions. The proposed AMM and its distributed realizations generalize a number of well-noted methods in the literature. Moreover, the theoretical convergence results of AMM are no worse than and sometimes even better than those of such existing specializations of AMM in the sense of rate order, problem assumption, etc. The generality of AMM provides insights into the design and analysis of distributed optimization algorithms and allows us to explore high-performance specializations of AMM when addressing specific convex optimization problems in practice.

APPENDIX

A. Proof of Lemma 1

Define \( g^{k+1} = - \nabla f (x^{k+1}) - \rho H x^{k+1} - \tilde{H}^{\ast} v^{k} \) and \( g^{\ast} = - \nabla f (x^{\ast}) - \tilde{H}^{\ast} v^{\ast} \). Due to (6) and (7), \( g^{k+1} \in \partial h (x^{k+1}) \) and \( g^{\ast} \in \partial h (x^{\ast}) \). It follows from (33) that

\[
\tilde{H}^{\ast} (v^{\ast} - v^{k}) = \nabla f (x^{k}) - \nabla f (x^{\ast}) + A (x^{k+1} - x^{k}) + g^{k+1} - g^{\ast} + \rho H x^{k+1}.
\]

Also, since \( \text{Null}(\tilde{H}^{\ast}) = \text{Null}(\tilde{H}) = S \), we have \( \tilde{H}^{\ast} x^{\ast} = 0 \).

This, along with (5), gives \( v^{k} - v^{k+1} = - \rho H x^{k+1} - x^{\ast} \). Thus, \( \langle v^{k} - v^{k+1}, v^{k} - v^{\ast} \rangle = - \rho \tilde{H}^{\ast} (x^{k+1} - x^{\ast}) \). \( v^{k} - v^{\ast} = \rho (x^{k+1} - x^{\ast}, \tilde{H}^{\ast} (v^{\ast} - v^{k})) \).

By substituting (50) into the above equation and using \( H x^{\ast} = 0 \), we obtain \( \langle v^{k} - v^{k+1}, v^{k} - v^{\ast} \rangle = \rho (x^{k+1} - x^{\ast}, \nabla f (x^{k}) - \nabla f (x^{\ast})) + \rho (x^{k+1} - x^{\ast}, g^{k+1} - g^{\ast}) + \rho (x^{k+1} - x^{\ast}, A (x^{k+1} - x^{\ast})) + \rho^{2} \| x^{k+1} \|^{2} \| h \| \).

In this equation, \( \langle x^{k+1} - x^{\ast}, g^{k+1} - g^{\ast} \rangle \geq 0 \), because \( g^{k+1} \in \partial h (x^{k+1}) \), \( g^{\ast} \in \partial h (x^{\ast}) \), and \( h \) is convex. Moreover, due to \( H \geq \tilde{H} \) and (5), we have \( \rho^{2} \| x^{k+1} \|^{2} \| H \| \geq \rho^{2} \| x^{k+1} \|^{2} \| \tilde{H} \| = \| v^{k+1} - v^{k} \|^{2} = \| v^{k} - v^{k+1} \|^{2} - (v^{k} - v^{k+1}, v^{k} - v^{\ast}) \). It follows that \( \frac{1}{\rho} (v^{k} - v^{k+1}, v^{k+1} - v^{\ast}) \geq \langle x^{k+1} - x^{\ast}, \nabla f (x^{k}) - \nabla f (x^{\ast}) + (x^{k+1} - x^{\ast}, A (x^{k+1} - x^{\ast})) \).

Hence, to prove (34), it suffices to
show that for any \( \eta \in (0, 1) \),
\[
(x^{k+1} - x^*, \nabla f(x^k) - \nabla f(x^*)) \geq \eta(x^k - x^*, \nabla f(x^k) - \nabla f(x^*)) - \frac{\|x^{k+1} - x^k\|_A^2}{4(1 - \eta)} . \tag{51}
\]
To do so, note from the AM-GM inequality and the Lipschitz continuity of \( \nabla f_i \) that for any \( i \in V \) and \( c_i > 0 \),
\[
\langle x^{k+1} - x_k, \nabla f_i(x^k) - \nabla f_i(x^*) \rangle \geq -c_i\|\|x^k_i\| - \nabla f_i(x^*)\|_V^2 \geq -c_i\|M_i(x^k - x^*) + \nabla f_i(x^k) - \nabla f_i(x^*)\|_V^2 .
\]
By adding the above inequality over all \( i \in V \) with \( c_i = \frac{1}{M\eta} \), we have \( \langle x^{k+1} - x_k, \nabla f(x^k) - \nabla f(x^*) \rangle + (1 - \eta)\langle x^k - x^*, \nabla f(x^k) - \nabla f(x^*) \rangle \geq -\frac{\|x^{k+1} - x_k\|_A^2}{4(1 - \eta)} . \]
Then, adding \( \eta(x^k - x^*, \nabla f(x^k) - \nabla f(x^*)) \) to both sides of this inequality leads to (51).

**B. Proof of Lemma 2**

Let \( k \geq 1 \). Due to (5), \( \hat{H} \hat{H}^\dagger x^k = \frac{1}{k} \sum_{i=1}^k \hat{H} \hat{H}^\dagger x^k = \frac{1}{k} \hat{H} \hat{H}^\dagger x^t \), so that (36) holds. Next, we prove (37). For simplicity, define \( J_i(x) = \langle \nabla f_i(x), x \rangle + \|x - x_i\|_A^2/2 + h(x) + \rho\|x\|^2_H/2 + (\nabla^T x) \begin{bmatrix} \rho & 0 \\ 0 & \hat{H} \hat{H} \end{bmatrix} x \forall x \geq 0 \). Since \( J_i(x) = \frac{\|x_i\|^2}{2} \) is convex, \( \langle J_i(x^t) - \frac{\|x_i\|^2}{2}, x - x^t \rangle \geq \langle (J_i(x^t) - \frac{\|x_i\|^2}{2} \rangle = \langle J_i(x^t) - \frac{\|x_i\|^2}{2}, x - x^t \rangle \). Due to (4), \( 0 \in \partial J_i(x^t) \). Thus, letting \( x^t = 0 \) in the above inequality, we obtain
\[
J_i(x^t) - J_i(x^t) \leq \langle x^t - x^t_i, x \rangle - \frac{\|x_i\|^2}{2} . \tag{52}
\]
In addition, because of (5) and \( H \geq \hat{H} \),
\[
\langle \hat{H} \hat{H}^\dagger x^t, x \rangle = \frac{1}{\rho} \langle \nabla^T \hat{H} x^t, x \rangle = \langle \|x^t\|^2 - \|v^T\|^2 - \|v^{t+1}\|^2 - \|v^t\|^2/2 \rangle/(2\rho) \]
\[
= \langle \|x^t\|^2 - \|v^T\|^2/2 \rangle/(2\rho) = -\|v^T\|^2/2 \rangle/(2\rho) = \|
\]
\[
\|x^t + v^T - v^t - v^{t+1} - v^t\|^2/2 \rangle/(2\rho) = \|
\]
\[
\leq \langle x^t - x^t_i, x \rangle - \frac{\|x_i\|^2}{2} \tag{53}
\]
Also, due to the convexity and the \( M_i \)-smoothness of each \( f_i \),
\[
f(x^{t+1}) - f(x^*) \geq \langle \nabla f_i(x^t), x^{t+1} - x^t \rangle + \|x^{t+1} - x^t\|^2/2 + (\nabla f_i(x^t), x^t - x^t) \]
\[
= \langle \nabla f_i(x^t), x^{t+1} - x^t \rangle + \|x^{t+1} - x^t\|^2/2 \tag{54}
\]
Through combining (52), (53), and (54) and utilizing \( Hx^k = 0 \), we derive
\[
f(x^{t+1}) + h(x^{t+1}) - h(x^*) \geq \|x^{t+1}\|^2 - \|v^t\|^2/2 \]
\[
\leq \frac{\|x^t - x^t_i\|^2}{2} - \frac{\|x^{t+1} - x^t\|^2}{2} + \frac{\|x^{t+1} - x^t\|^2}{2} \tag{55}
\]
Now adding (55) over \( t = 0, \ldots, k - 1 \) yields \( \sum_{i=1}^k \sum_{t=0}^{k-1} \frac{\|x^t - x^t_i\|^2}{2} + \frac{\|x^{t+1} - x^t\|^2}{2} \). Moreover, the convexity of \( f \) and \( h \) implies \( f(\bar{x}^k) + h(\bar{x}^k) \leq \frac{1}{k} \sum_{i=1}^k (f(x^i) + h(x^i)) \). Combining the above results in (37).

Finally, to prove (38), note that \( x^* \) is an optimum of \( \min_{x, \hat{x}} f(x) + h(\hat{x}) + (v^*, H \hat{x}) \). It follows from \( H \hat{x} = 0 \) that
\[
f(x^*) + h(\bar{x}^*) \leq f(\bar{x}^t) + h(\bar{x}^t) + (v^*, H \bar{x}^t) \leq f(\bar{x}^t) + h(\bar{x}^t) + \|v^*\| \|H \bar{x}^t\| . \]
This and (36) imply (38).

**C. Proof of Lemma 3**

From the definition of \( G \), for each \( t \geq 0 \),
\[
\|z^t - z^*\|^2_{G'} - \|z^{t+1} - z^*\|^2_G = 2(G(z^t - z^t+1), z^t+1 - z^*)
\]
\[
= 2(A(x^t - x^{t+1}), x^{t+1} - x^t) + \frac{2}{\rho} (v^t - v^{t+1}, v^{t+1} - v^*) . \tag{56}
\]
By substituting (34) with \( A^t = A \) and \( \eta = 0 \) into (56),
\[
\|z^t - z^*\|_G^2 - \|z^{t+1} - z^*\|_G^2 \geq \frac{1}{\rho} (\|v^t - v^{t+1}\|^2 + \|v^{t+1} - v^*\|^2)
\]
\[
+ \|x^{t+1} - x^t\|^2_{A} \geq (1 - \frac{\rho}{A}) \|x^t - x^t\|^2_{A} \tag{57}
\]
where the last step is due to \( \sigma A \leq A \). Adding (57) over \( t = 0, \ldots, k - 1 \) and using \( \|v^T - v^t\|^2 \leq \|v^T - v^t\|^2 \), we have
\[
\frac{1}{\rho} (\|v^T - v^t\|^2 + (1 - \frac{\rho}{A}) \sum_{t=0}^{k-1} \|x^{t+1} - x^t\|^2_{A} \leq \|v^t - v^*\|^2 . \]
Therefore, (39) can be proved by combining the above inequality with \( \|v^t - v^t\|^2 \leq \|v^T - v^*\|^2 + \|v^T - v^*\|^2 . \) Furthermore, the above inequality, along with \( A \geq \rho \), implies (40).
On the other hand, since \( A_{\ell} \leq A_k \leq A_{\ast} \), we have \( \| A_k - A_{\ast} \| \leq \Delta / 2 \). Due to (41), there exist \( \beta > 0 \) and \( \sigma \in (0, 1) \) such that (44) holds. Then, through the AM-GM inequality, \((x_k^1 - x^\ast)(A_k - A_{\ast})(x_k^1 - x^\ast)\) = \((x_k^1 - x^\ast)(A_k - A_{\ast})(x_k^1 - x^\ast)\) + \((x_k^1 - x^\ast)(A_k - A_{\ast})(x_k^1 - x^\ast)\) \(\geq -\frac{1}{2}(\beta \| x_k^1 - x^\ast \|^2 + \delta A_{\ast})\| x_k^1 - x^\ast \|^2 - \frac{1}{2}\| x_k^1 - x^\ast \|^2\), which further implies

\[
\| x_k^1 - x^\ast \|^2_{A_{\ast}} + 2\| x_k^1 - x^\ast \|^2 + (A_k - A_{\ast})(x_k^1 - x^\ast) \geq \frac{1}{2}(\beta \| x_k^1 - x^\ast \|^2 + \delta A_{\ast})\| x_k^1 - x^\ast \|^2.
\]

Substituting this into (59) and applying (5), we obtain

\[
c - c_k^1 + \geq (1 - \sigma)\| x_k^1 - x^\ast \|^2_{A_{\ast}} - \beta \delta \| x_k^1 - x^\ast \|^2 + 2\sigma \| x_k^1 - x^\ast \|^2 + \rho \| x_k^1 - x^\ast \|^2_{A_{\ast}}.
\]

Due to the hypothesis \( c_k \leq c^0 \), we have \( x_k^1 \in C \). It then follows from Proposition 2 and \( H^\ast x \geq 0 \) that \( 2\| x_k^1 - x^\ast \|^2 + \| f(x_k^1) - \nabla f(x^\ast) \|^2 \geq 2\| x_k^1 - x^\ast \|^2 + \rho \| x_k^1 - x^\ast \|^2_{A_{\ast}} \). This and (60) give

\[
c_k - c_k^1 \geq (1 - \sigma)\| x_k^1 - x^\ast \|^2_{A_{\ast}} - \beta \delta \| x_k^1 - x^\ast \|^2 + 2\sigma \| x_k^1 - x^\ast \|^2 + \rho \| x_k^1 - x^\ast \|^2_{A_{\ast}}.
\]

Note that \( \sigma \in (0, 1) \), \( \beta \delta < 2\sigma \rho \nu \) due to (44), and \( \eta \in (0, 1) \). Thus, the right-hand side of (61) is nonnegative, which implies \( c_k \leq c^0 \).

\[\text{G. Proof of Theorem 3}\]

In the proof of Lemma 4, we have shown that \((c_k)_{k=0}^\infty\) is nonincreasing. Thus, \( \| z_k^1 - x^\ast \|^2_{L_j} \leq c_k \leq c^0 = \frac{d_2^2}{\rho} \). It follows that \( \| z_k^1 - x^\ast \|^2 \leq \| z_k^1 - x^\ast \|^2 + \| x^\ast - x_k^1 \|^2 \leq \| z_k^1 - x^\ast \|^2 + \| x^\ast - x_k^1 \|^2 = \| z_k^1 - x^\ast \|^2_{L_j} \). From (44) that \( 2\sigma \rho \nu - \beta \delta > 0 \) and \( A_{\ast} \geq 0 \). Also note that \( \theta_0 > 0, \theta_2 > 0, \rho > 0, \lambda_1 > 0, \eta \in (0, 1) \), \( \bar{H} \geq O \). Range(\( \bar{H} \)) = Range(\( H \)) = \( S^1 \), and \( \sigma \in (0, 1) \). Therefore, there exists \( \delta \in (0, 1) \) such that \( B(\delta) \geq O \forall \xi \in \{ 1, 2, 3 \} \), which guarantees (43).

\[\text{REFERENCES}\]

[1] A. Nedić, “Convergence rate of distributed averaging dynamics and optimization in networks,” Found. Trends Syst. Control, vol. 2, no. 1, pp. 1–100, 2015.
[2] D. Bajočić, D. Jeković, N. Krejić, and N. K. Jerińik, “Newton-like method with diagonal correction for distributed optimization,” SIAM J. Optim., vol. 27, no. 2, pp. 1171–1203, 2017.
[3] F. Mansoori and E. Wei, “Fast distributed asynchronous Newton-based optimization algorithm,” IEEE Trans. Autom. Control, vol. 65, no. 7, pp. 2769–2784, Jul. 2020.
[4] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, “On the linear convergence of the ADMM in decentralized consensus optimization,” IEEE Trans. Signal Process., vol. 62, no. 7, pp. 1750–1761, Apr. 2014.
[5] W. Shi, Q. Ling, G. Wu, and W. Yin, “EXTRA: An exact first-order algorithm for decentralized consensus optimization,” SIAM J. Optim., vol. 25, no. 2, pp. 944–966, 2015.
[6] Y. Liu, W. Xu, G. Wu, Z. Tian, and Q. Ling, “Communication-censored ADMM for decentralized consensus optimization,” IEEE Trans. Signal Process., vol. 67, no. 10, pp. 2565–2579, May 2019.

[7] Z. Li, W. Shi, and M. Yan, “A decentralized proximal-gradient method with network independent stepsizes and separated convergence rates,” IEEE Trans. Signal Process., vol. 67, no. 17, pp. 4494–4506, Sep. 2019.

[8] I. Nariotnicola and G. Notarstefano, “Asynchronous distributed optimization via randomized dual proximal gradient,” IEEE Trans. Autom. Control, vol. 62, no. 5, pp. 2095–2106, May 2017.

[9] A. Mahdoumi and A. Ozdaglar, “Convergence rate of distributed ADMM over networks,” IEEE Trans. Autom. Control, vol. 62, no. 10, pp. 5082–5095, Oct. 2017.

[10] J. Zeng, T. He, and M. Wang, “A fast proximal gradient algorithm for decentralized composite optimization over directed networks,” Syst. Control Lett., vol. 107, no. 2017, pp. 36–43, 2017.

[11] N. S. Aybat, Z. Wang, T. Lin, and S. Ma, “Distributed linearized alternating direction method of multipliers for composite convex consensus optimization,” IEEE Trans. Autom. Control, vol. 63, no. 1, pp. 5–20, Jan. 2018.

[12] W. Shi, Q. Ling, G. Wu, and W. Yin, “A proximal gradient algorithm for decentralized composite optimization,” IEEE Trans. Signal Process., vol. 63, no. 22, pp. 6013–6023, Nov. 2015.

[13] J. Xu, S. Zhu, Y. Soh, and L. Xie, “A Bregman splitting scheme for distributed optimization over networks,” IEEE Trans. Autom. Control, vol. 61, no. 10, pp. 2947–2957, Oct. 2016.

[14] J. Lei, H. Chen, and H. Fang, “Primal-dual algorithm for distributed constrained optimization,” Syst. Control Lett., vol. 96, pp. 110–117, 2016.

[15] X. Wu and J. Lu, “Fenchel dual gradient methods for distributed convex optimization over time-varying networks,” IEEE Trans. Autom. Control, vol. 64, no. 11, pp. 4629–4636, Nov. 2019.

[16] M. Hong and T. Chang, “Stochastic proximal gradient consensus over random networks,” IEEE Trans. Signal Process., vol. 65, no. 11, pp. 2933–2948, Jun. 2017.

[17] P. Bianchi, W. Hachem, and F. fetusler, “A coordinate descent primal-dual algorithm and application to distributed asynchronous optimization,” IEEE Trans. Autom. Control, vol. 61, no. 10, pp. 2947–2957, Oct. 2016.

[18] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, “A decentralized second-order method with exact linear convergence rate for consensus optimization,” IEEE Trans. Signal Inf. Process. Netw., vol. 2, no. 4, pp. 507–522, Dec. 2016.

[19] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, “DQM: Decentralized quadratically approximated alternating direction method of multipliers,” IEEE Trans. Signal Process., vol. 64, no. 19, pp. 5158–5173, Oct. 2016.

[20] X. Wu, Z. Qu, and J. Lu, “A second-order proximal algorithm for consensus optimization,” IEEE Trans. Autom. Control, vol. 66, no. 4, pp. 1864–1871, Apr. 2021.

[21] J. Xu, Y. Tian, Y. Sun, and G. Scutari, “Distributed algorithms for composite optimization: Unified framework and convergence analysis,” IEEE Trans. Signal Process., vol. 69, pp. 3555–3570, Jun. 2021.

[22] S. A. Alghunaim, E. Ryu, K. Yuan, and A. H. Sayed, “Decentralized proximal gradient algorithms with linear convergence rates,” IEEE Trans. Autom. Control, vol. 66, no. 6, pp. 2787–2794, Jun. 2021.

[23] K. Yuan, Q. Ling, and W. Yin, “On the convergence of decentralized gradient descent,” SIAM J. Optim., vol. 26, no. 3, pp. 1835–1854, 2016.

[24] D. Yuan, Y. Hong, D. W. C. Ho, and C. Jiang, “Optimal distributed stochastic mirror descent for strongly convex optimization,” Automatica, vol. 90, pp. 196–203, 2018.

[25] A. Nedić, A. Olshevsky, and W. Shi, “Achieving geometric convergence for distributed optimization over time-varying graphs,” SIAM J. Optim., vol. 27, no. 4, pp. 2597–2633, 2017.

[26] G. Qiu and N. Li, “Harnessing smoothness to accelerate distributed optimization,” IEEE Trans. Control Netw. Syst., vol. 5, no. 3, pp. 1245–1260, Sep. 2017.

[27] S. Pu, W. Shi, J. Xu, and A. Nedić, “Push-pull gradient methods for distributed optimization in networks,” IEEE Trans. Autom. Control, vol. 66, no. 1, pp. 1–16, Jan. 2021.

[28] P. D. Lorenzo and G. Scutari, “NEXT: In-network nonconvex optimization,” IEEE Trans. Signal Inf. Process. Netw., vol. 2, no. 2, pp. 120–136, Jun. 2016.

[29] A. Daneshmand, Y. Sun, G. Scutari, F. Facchinei, and B. Sadler, “Decentralized dictionary learning over time-varying digraphs,” J. Mach. Learn. Res., vol. 20, pp. 1–62, 2019.

[30] G. Scutari and Y. Sun, “Distributed nonconvex constrained optimization over time-varying digraphs,” Math. Program., vol. 176, pp. 497–544, 2016.

[31] I. Nariotnicola, Y. Sun, G. Scutari, and G. Notarstefano, “Distributed big-data optimization via blockwise gradient tracking,” IEEE Trans. Autom. Control, vol. 66, no. 5, pp. 2045–2060, May 2021.

[32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” Found. Trends Mach. Learn., vol. 3, no. 1, pp. 1–122, 2011.

[33] S. Kakade, S. Shalev-Shwartz, and A. Tewari, “On the duality of strong convexity and strong smoothness: Learning applications and matrix regularization,” Toyota Technol. Inst. Chinitgo, Tech. Rep., 2009.

[34] A. Beck and M. Teboulle, “Mirror descent and nonlinear projected subgradient methods for convex optimization,” Oper. Res. Lett., vol. 31, no. 3, pp. 167–175, 2003.

[35] M. Spivak, Calculus on Manifolds: A Modern Approach to Classical Theorems of Advanced Calculus. Reading, MA, USA: Addison-Wesley, 1965.

[36] F. Bach, “Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression,” J. Mach. Learn. Res., vol. 15, pp. 595–627, 2014.

[37] R. Xin, U. A. Khan, and S. Kar, “An improved convergence analysis for decentralized online stochastic non-convex optimization,” IEEE Trans. Signal Process., vol. 69, pp. 1842–1858, Mar. 2021.

Xuyang Wu received the B.S. degree in information and computing science from Northwestern Polytechnical University, Xi’an, China, in 2015, and the Ph.D. degree in communication and information systems from the School of Information Science and Technology, ShanghaiTech University, Shanghai, China, in 2020. He is currently a Postdoctoral Researcher with the Division of Decision and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden. His research interests include distributed optimization, large-scale optimization, and their applications in IoT and power systems.

Jie Lu (Member, IEEE) received the B.S. degree in information engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007, and the Ph.D. degree in electrical and computer engineering from the University of Oklahoma, Norman, OK, USA, in 2011. She is currently an Associate Professor with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China. Before she joined ShanghaiTech University in 2015, she was a Postdoctoral Researcher with the KTH Royal Institute of Technology, Stockholm, Sweden, and with the Chalmers University of Technology, Gothenburg, Sweden from 2012 to 2015. Her research interests include distributed optimization, optimization theory and algorithms, and networked dynamical systems.