A Communication-Efficient Decentralized Newton's Method with Provably Faster Convergence

Huikang Liu, Jiaojiao Zhang, Anthony Man-Cho So, and Qing Ling

***Abstract*—In this paper, we consider a strongly convex finite-sum minimization problem over a decentralized network and propose a communication-efficient decentralized Newton's method for solving it. The main challenges in designing such an algorithm come from three aspects: (i) mismatch between local gradients/Hessians and the global ones; (ii) cost of sharing second-order information; (iii) tradeoff among computation and communication. To handle these challenges, we first apply dynamic average consensus (DAC) so that each node is able to use a local gradient approximation and a local Hessian approximation to track the global gradient and Hessian, respectively. Second, since exchanging Hessian approximations is far from communication-efficient, we require the nodes to exchange the compressed ones instead and then apply an error compensation mechanism to correct for the compression noise. Third, we introduce multi-step consensus for exchanging local variables and local gradient approximations to balance between computation and communication. With novel analysis, we establish the globally linear (resp., asymptotically super-linear) convergence rate of the proposed algorithm when m is constant (resp., tends to infinity), where $m \geq 1$ is the number of consensus inner steps. To the best of our knowledge, this is the first super-linear convergence result for a communication-efficient decentralized Newton's method. Moreover, the rate we establish is provably faster than those of first-order methods. Our numerical results on various applications corroborate the theoretical findings.**

***Index Terms*—Decentralized optimization, convergence rate, Newton's method, compressed communication**

I. Introduction

In this paper, we consider solving a finite-sum optimization problem defined over an undirected, connected network with n nodes:

$$x^* = \arg\min_{x\in\mathbb{R}^d} F(x) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(x). \tag{1}$$

Here, $x \in \mathbb{R}^d$ is the decision variable and $f_i : \mathbb{R}^d \to \mathbb{R}$ is a twice-continuously differentiable function privately owned by node i. The entire objective function F is assumed to be strongly convex. Each node is allowed to exchange limited information with its neighbors during the optimization process. To make (1) separable across the nodes, one common way is to introduce a local copy $x_i \in \mathbb{R}^d$ of x for node i and then force all the local copies to be equal by adding consensus constraints. This leads to the following alternative formulation of Problem (1):

$$\begin{aligned}\mathbf{x}^* = \arg\min_{\{x_i\}_{i=1}^n} \ & \frac{1}{n}\sum_{i=1}^{n} f_i(x_i) \\ \text{s.t.} \ \ & x_i = x_j, \ \forall j \in \mathcal{N}_i, \ \forall i.\end{aligned} \tag{2}$$

Here, $\mathbf{x}^* \triangleq [x^*; \ldots; x^*] \in \mathbb{R}^{nd}$ and $\mathcal{N}_i$ is the set of neighbors of node i. The equivalence between (1) and (2) holds when the network is connected. Decentralized optimization problems in the form of (2) appear in various applications, such as federated learning [1], sensor networking [2], statistical learning [3], etc.

Decentralized algorithms for solving (2) are well studied. All nodes cooperatively obtain the common optimal solution x^*, simultaneously minimizing the objective function and reaching consensus. Generally speaking, minimization is realized by inexact descent on local objective functions and consensus is realized by variable averaging with a mixing matrix [4]. Below, we briefly review the existing first-order and second-order decentralized algorithms for solving (2).

A. Decentralized First-order Methods

First-order methods enjoy low per-iteration computational complexity and thus are popular. Decentralized gradient descent (DGD) is studied in [5], [6], where each node updates its local copy by a weighted average step on local copies from its neighbors, followed by a minimization step along its local gradient descent direction. With a fixed step size, DGD only converges to a neighborhood of x^*. This disadvantage can in part be explained by the observation that the local gradient is generally not a satisfactory estimate of the global one, even though the local copies are all equal to the optimal solution x^*. To construct a better local direction, various works with bias-correction techniques are proposed, such as primal-dual [7]–[10], exact diffusion [11], and gradient tracking [12]–[14]. For example, gradient tracking replaces the local gradient in DGD with a local gradient approximation obtained by the dynamic average consensus (DAC) technique, which leads to exact convergence with a fixed step size. A general decentralized heavy-ball method, which includes several accelerated first-order methods as special cases, is presented in [15]. Unified

Huikang Liu is with the Research Institute for Interdisciplinary Sciences, Shanghai University of Finance and Economics (e-mail: liuhuikang@sufe.edu.cn). Jiaojiao Zhang is with the Division of Decision and Control Systems, KTH Royal Institute of Technology (e-mail: jiaoz@kth.se). Anthony Man-Cho So is with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong (e-mail: manchoso@se.cuhk.edu.hk). Qing Ling is with the School of Computer Science and Engineering and Guangdong Provincial Key Laboratory of Computational Science, Sun Yat-Sen University, and also with the Pazhou Lab (e-mail: lingqing556@mail.sysu.edu.cn).

Huikang Liu is supported by National Natural Science Foundation of China grant 72192832. Qing Ling is supported by National Natural Science Foundation of China grants 61973324 and 12126610, Guangdong Basic and Applied Basic Research Foundation grant 2021B1515020094, and Guangdong Provincial Key Laboratory of Computational Science grant 2020B1212060032.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes additional mathematical derivations.

frameworks for first-order algorithms are investigated in [16], [17].

In the centralized setting, it is well-known that the convergence of first-order algorithms suffers from dependence on κ_F, the condition number of the objective function F. In the decentralized setting, the dependence is not only on κ_F but also on the network. Specifically, let σ be the second largest singular value of the mixing matrix used in decentralized optimization and $\frac{1}{1-\sigma}$ be the condition number of the underlying communication graph. A network with larger $\frac{1}{1-\sigma}$ has a weaker information diffusion ability. For strongly convex and smooth problems, the work [18] establishes the lower bounds $O\left(\sqrt{\kappa_F}\log\frac{1}{\epsilon}\right)$ and $O\left(\sqrt{\frac{\kappa_F}{1-\sigma}}\log\frac{1}{\epsilon}\right)$ on the computation and communication costs for decentralized first-order algorithms to reach an ϵ-optimal solution, respectively. The lower bounds are achieved or nearly achieved in [19], [20], where multi-step consensus is introduced to balance the computation and communication costs.

B. Decentralized Second-order Methods

In the centralized setting, Newton's method is proved to have a locally quadratic convergence rate that is independent of κ_F. However, whether there is a communication-efficient decentralized variant of Newton's method with κ_F-independent super-linear convergence rate under mild assumptions is still an open question. On the one hand, some decentralized second-order methods have provably faster rates but suffer from inexact convergence, high communication cost, or requiring strict assumptions. The work [21] extends the network Newton's method in [22] for minimizing a penalized approximation of (1) and shows that the convergence rate is super-linear in a specific neighborhood near the optimal solution of the penalized problem. Beyond this neighborhood, the rate becomes linear. The work [23] proposes an approximate Newton's method for the dual problem of (2) and establishes a super-linear convergence rate within a neighborhood of the primal-dual optimal solution. However, in each iteration, it needs to solve the primal problem exactly to obtain the dual gradient and call a solver to obtain the local Newton direction. The work [24] proposes a decentralized adaptive Newton's method, which uses the communication-inefficient flooding technique to make the global gradient and Hessian available to each node. In this way, each node conducts exactly the same update so that the global super-linear convergence rate of the centralized Newton's method with Polyak's adaptive step size still holds. The work [25] proposes a decentralized Newton-type method with cubic regularization and proves faster convergence up to statistical error under the assumption that each local Hessian is close enough to the global Hessian. The work [26] studies quadratic local objective functions and shows that for a distributed Newton's method, the computation complexity depends only logarithmically on κ_F with the help of exchanging the entire Hessian matrices. The algorithm in [26] is close to that in [27], but the latter has no convergence rate guarantee.

On the other hand, some works are devoted to developing efficient decentralized second-order algorithms with similar computation and communication costs per iteration to first-order algorithms. However, these methods only have globally linear convergence rate, which is no better than that of first-order methods [28]–[34]. Here we summarize several reasons for the lack of provably faster rates: (i) The information fusion over the network is realized by averaging consensus, whose convergence rate is at most linear [4]. (ii) The global Hessian is estimated just from local Hessians [28]–[32] or from Hessian inverse approximations constructed with local gradient approximations [33], [34]. The purpose is to avoid the communication of entire Hessian matrices, but a downside is that the nodes are unable to fully utilize the global second-order information. (iii) The global Hessian matrices are typically assumed to be uniformly bounded, which simplifies the analysis but leads to under-utilization of the curvature information [28]–[34]. (iv) For the centralized Newton's method, backtracking line search is vital for convergence analysis. It adaptively gives a small step size at the early stage to guarantee global convergence with arbitrary initialization and always gives a unit step size after reaching a neighborhood of the optimal solution to guarantee locally quadratic convergence rate. However, backtracking line search is not affordable in the decentralized setting since it is expensive for all the nodes to jointly calculate the global objective function value.

To the best of our knowledge, there is no decentralized Newton's method that, under mild assumptions, is not only communication-efficient but also inherits the κ_F-independent super-linear convergence rate of the centralized Newton's method. Therefore, in this paper we aim to address the following question: *Can we design a communication-efficient decentralized Newton's method that has a provably κ_F-independent super-linear convergence rate?*

C. Major Contributions

To answer these questions, we propose a decentralized Newton's method with multi-step consensus and compression and establish its convergence rate. Roughly speaking, our method proceeds as follows. In each iteration, each node moves one step along a local approximated Newton direction, followed by variable averaging to improve consensus. To construct the local approximated Newton direction, we use the DAC technique to obtain a gradient approximation and a Hessian approximation, which track the global gradient and global Hessian, respectively. To avoid having each node to transmit the entire local Hessian approximation, we design a compression procedure with error compensation to estimate the global Hessian in a communication-efficient way. In other words, each node is able to obtain more accurate curvature information by exchanging the compressed local Hessian approximations with its neighbors, without incurring a high communication cost. In addition, to balance between computation and communication costs, we use multi-step consensus for communicating the local copies of the decision variable and the local gradient approximations. Multi-step consensus helps to obtain not only a globally linear rate that is independent of the graph but also a faster local convergence rate.

Theoretically, we show, with novel analysis, that our proposed method enjoys a provably faster convergence rate than

those of decentralized first-order methods. The convergence process is split into two stages. In Stage I, we use a small step size and get globally linear convergence at the contraction rate of $1-O\left(\frac{1}{\kappa_F}\right)\min\left\{\frac{(1-\sigma^2)^3}{\sigma^{m-1}},\frac{1}{2}\right\}$ with arbitrary initialization. Here, $\frac{1}{1-\sigma}$ is the condition number of the graph, κ_F is the condition number of the objective function, and m is the number of consensus inner steps. This globally linear rate holds for any $m\geq 1$. As a special case, when $m\geq\frac{\log 2(1-\sigma^2)^3}{\log\sigma}+1$, the contraction rate in Stage I becomes $1-O\left(\frac{1}{\kappa_F}\right)$, which is independent of the graph. When the local copies are close enough to the optimal solution, the algorithm enters Stage II, where we use a unit step size and get the faster convergence rate of $\sigma^{\frac{m}{2}}$. This implies that we have a κ_F-independent linear rate when m is a constant and an asymptotically super-linear rate when m increases to infinity as the number of iterations increases to infinity. When $m>\frac{4\log(4\kappa_F)}{-\log\sigma}$, the communication complexity in Stage II is $O\left(\frac{1}{-\log\sigma}\log\frac{1}{\epsilon}\right)$. Since Stage I terminates within a finite number of iterations, when ϵ is small, our algorithm, albeit using multi-step consensus, still has a lower total communication cost than first-order algorithms due to the independence of κ_F in Stage II. A comparison of the iteration complexity of existing decentralized first-order and second-order methods are summarized in Table I.

TABLE I: Iteration complexity to reach an ϵ-optimal solution for decentralized consensus optimization algorithms

Algorithm	Iteration complexity
DLM [9]	$O\left(\max\left\{\frac{\kappa_F^2\lambda_{\max}(\mathcal{L}_u)}{\lambda_{\min}(\mathcal{L}_u)},\frac{(\lambda_{\max}(\mathcal{L}_u))^2}{\lambda_{\min}(\mathcal{L}_u)\hat{\lambda}_{\min}(\mathcal{L}_o)}\right\}\log\frac{1}{\epsilon}\right)$ [1]
EXTRA [8]	$O\left(\frac{\kappa_F^2}{1-\sigma}\log\frac{1}{\epsilon}\right)$ [2]
GT [12]	$O\left(\frac{\kappa_F^2}{(1-\sigma)^2}\log\frac{1}{\epsilon}\right)$
DQM [28]	$O\left(\max\left\{\left(\frac{\lambda_{\max}(\mathcal{L}_u)}{\hat{\lambda}_{\min}(\mathcal{L}_o)}\right)^2,\frac{\kappa_F\lambda_{\max}(\mathcal{L}_u)}{\hat{\lambda}_{\min}(\mathcal{L}_o)}\right\}\log\frac{1}{\epsilon}\right)$ [3]
ESOM [35]	$O\left(\frac{\kappa_F^2}{\hat{\lambda}_{\min}(I_n-W)}\log\frac{1}{\epsilon}\right)$ [4]
NT [31]	$O\left(\max\left\{\kappa_F^2+\kappa_F\sqrt{\kappa_g},\frac{\kappa_g^{3/2}}{\kappa_F}+\kappa_F\sqrt{\kappa_g}\right\}\log\frac{1}{\epsilon}\right)$ [5]
This Paper	Stage I: $\min\left\{K,O\left(\kappa_F\max\left\{\frac{\sigma^{m-1}}{(1-\sigma^2)^3},2\right\}\log\frac{1}{\epsilon}\right)\right\}$ [6]
	Stage II: $O\left(\frac{1}{-m\log\sigma}\log\frac{1}{\epsilon}\right)$ [7]

Notation. We use I_d to denote the $d\times d$ identity matrix, 1_n to denote the n-dimensional column vector of all ones, $\|\cdot\|$ to denote the Euclidean norm of a vector or the largest singular value of a matrix, $\|\cdot\|_F$ to denote the Frobenius norm, and $\otimes$ to denote the Kronecker product. For a matrix A, we use $A\geq 0$ to indicate that each entry of A is non-negative. For a symmetric matrix A, we use $A\succeq 0$ and $A\succ 0$ to indicate that A is positive semidefinite and positive definite, respectively. For two matrices A and B of the same dimensions, we use $A\geq B$, $A\succeq B$, and $A\succ B$ to indicate that $A-B\geq 0$, $A-B\succeq 0$, and $A-B\succ 0$, respectively. We use $\lambda_{\max}(\cdot)$, $\lambda_{\min}(\cdot)$, and $\hat{\lambda}_{\min}(\cdot)$ to denote the largest, smallest, and the smallest positive eigenvalues of a matrix, respectively.

For $x_1,\ldots,x_n\in\mathbb{R}^d$, we define the aggregated variable $\mathbf{x}=[x_1;\ldots;x_n]\in\mathbb{R}^{nd}$. The aggregated variables $\mathbf{d}$ and $\mathbf{g}$ are defined similarly. We define the average variable over all the nodes at time step k as $\overline{\mathbf{x}}^k=\frac{1}{n}\sum_{i=1}^n x_i^k\in\mathbb{R}^d$. The average variables $\overline{\mathbf{d}}^k$ and $\overline{\mathbf{g}}^k$ are defined similarly. We define the aggregated gradient at time step k as $\nabla f(\mathbf{x}^k)=\left[\nabla f_1(x_1^k);\ldots;\nabla f_n(x_n^k)\right]\in\mathbb{R}^{nd}$, the average of all the local gradients at the local variables as $\overline{\nabla f}(\mathbf{x}^k)=\frac{1}{n}\sum_{i=1}^n\nabla f_i(x_i^k)\in\mathbb{R}^d$, and the average of all the local gradients at the common average $\overline{\mathbf{x}}^k$ as $\nabla F(\overline{\mathbf{x}}^k)=\frac{1}{n}\sum_{i=1}^n\nabla f_i(\overline{\mathbf{x}}^k)\in\mathbb{R}^d$. The aggregated Hessian $\nabla^2 f(\mathbf{x}^k)\in\mathbb{R}^{nd\times d}$, the average of all the local Hessian at the local variables $\overline{\nabla^2 f}(\mathbf{x}^k)\in\mathbb{R}^{d\times d}$, and the average of all the local Hessians at the common average $\nabla^2 F(\overline{\mathbf{x}}^k)\in\mathbb{R}^{d\times d}$ are defined similarly. Given the matrices $H_i^k\in\mathbb{R}^{d\times d}$, we define the aggregated matrix $\mathbf{H}^k=[H_1^k;\ldots;H_n^k]\in\mathbb{R}^{nd\times d}$. The aggregated matrices $\mathbf{E}^k$, $\tilde{\mathbf{H}}^k$, and $\hat{\mathbf{H}}^k$ are defined similarly. We define the average variable over all the nodes at time step k as $\overline{\mathbf{H}}^k=\frac{1}{n}\sum_{i=1}^n H_i^k$ and $\mathrm{diag}\{H_i\}\in\mathbb{R}^{nd\times nd}$ as the block diagonal matrix whose i-th block is $H_i\in\mathbb{R}^{d\times d}$. We define $\mathbf{W}=W\otimes I_d\in\mathbb{R}^{nd\times nd}$ and $\mathbf{W}^\infty=\frac{1_n1_n^T}{n}\otimes I_d\in\mathbb{R}^{nd\times nd}$.

II. Problem Setting and Algorithm Development

In this section, we give the problem setting and the basic assumptions. Then, we propose a decentralized Newton's method with multi-step consensus and compression.

A. Problem Setting

We consider an undirected, connected graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$, where $\mathcal{V}=\{1,\ldots,n\}$ is the set of nodes and $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ is the set of edges. We use $(i,j)\in\mathcal{E}$ to indicate that nodes i and j are neighbors, and neighbors are allowed to communicate with each other. We use $\mathcal{N}_i$ to denote the set of neighbors of node i and itself. We introduce a mixing matrix $W\in\mathbb{R}^{n\times n}$ to model the communication among nodes. The mixing matrix is assumed to satisfy the following:

Assumption 1. *The mixing matrix W is non-negative, symmetric, and doubly stochastic (i.e., $w_{ij}\geq 0$ for all $i,j\in\{1,\ldots,n\}$, $W=W^T$, and $W1_n=1_n$) with $w_{ij}=0$ if and only if $j\notin\mathcal{N}_i$.*

Assumption 1 implies that the null space of I_n-W is $\mathrm{span}(1_n)$, the eigenvalues of W lie in $(-1,1]$, and 1 is an eigenvalue of W of multiplicity 1. Let σ be the second largest singular value of W. Under Assumption 1, we have

$$\sigma=\|W-W^\infty\|<1.$$

[1] Here, $\mathcal{L}_u$ and $\mathcal{L}_o$ are the unoriented and oriented Laplacian defined in [9], respectively. The rate is obtained when $\alpha=\frac{L_1\kappa_F}{\lambda_{\min}(\mathcal{L}_u)}$ and $\epsilon=L_1\kappa_F$ with L_1 being the Lipschitz constant of the gradient.

[2] Here, W is the mixing matrix, $\tilde{W}=\frac{I_n+W}{2}$, and $\alpha=\frac{0.5\hat{\lambda}_{\min}(\tilde{W})}{L_1\kappa_F}$.

[3] Here, the convergence is local and $\alpha=\frac{L_1}{\lambda_{\max}(\mathcal{L}_u)\hat{\lambda}_{\min}(\mathcal{L}_0)}$.

[4] Here, the convergence is local and the number of consensus inner steps goes to infinity.

[5] Here, $\kappa_g=\frac{\lambda_{\max}(I_n-W)}{\hat{\lambda}_{\min}(I_n-W)}$ as defined in [31] and the convergence is local.

[6] Here, K is a finite constant defined in (18) and is independent of ϵ. Stage I terminates within K steps and Stage II is independent of κ_F. For simplicity, we use the local Lipschitz constant of the gradient L_1 to define κ_F. Actually, by following the techniques in [30], we can use the global Lipschitz constant of the gradient for L_1.

[7] Here, m is set as a constant.

Usually, σ is used to represent the connectedness of the graph [5], [8]. Mixing matrices satisfying Assumption 1 are frequently used in decentralized optimization over an undirected, connected network; see, e.g., [36] for details.

Throughout the paper, we make the following assumptions on the local objective functions.

Assumption 2. *Each f_i is twice-continuously differentiable. Both the gradient and Hessian are Lipschitz continuous, i.e.,*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_1\|x-y\|$$

and

$$\|\nabla^2 f_i(x) - \nabla^2 f_i(y)\| \leq L_2\|x-y\|$$

for all $x, y \in \mathbb{R}^d$, where $L_1 > 0$ and $L_2 > 0$ are the Lipschitz constants of the local gradient and local Hessian, respectively.

Assumption 3. *The entire objective F is μ-strongly convex for some constant $\mu > 0$, i.e.,*

$$\nabla^2 F(x) \succeq \mu I_d$$

for all $x \in \mathbb{R}^d$, where μ is the strong convexity constant.

We should remark that in Assumption 3, we only assume the entire objective function F to be strongly convex. The local objective function f_i on each node could even be nonconvex, which makes our analysis more general.

To avoid having each node to communicate the entire local Hessian approximation, we design a compression procedure with a deterministic contractive compression operator $\mathcal{Q}(\cdot)$ that satisfies the following assumption.

Assumption 4. *The deterministic contractive compression operator $\mathcal{Q} : \mathbb{R}^{d\times d} \to \mathbb{R}^{d\times d}$ satisfies*

$$\|\mathcal{Q}(A) - A\|_F \leq (1-\delta)\|A\|_F \tag{3}$$

for all $A \in \mathbb{R}^{d\times d}$, where $\delta \in (0, 1]$ is a constant determined by the compression operator.

We now present two concrete examples of such an operator. Let $A = \sum_{i=1}^{d} \sigma_i u_i v_i^T$ be the singular value decomposition of the matrix A, where σ_i is the i-th largest singular value with u_i and v_i being the corresponding singular vectors. The Rank-K compression operator outputs $\mathcal{Q}(A) = \sum_{i=1}^{K} \sigma_i u_i v_i^T$, which is a rank-$K$ approximation of A. For Top-K compression operator, $\mathcal{Q}(A)$ keeps the K largest entries (in terms of the absolute value) of the matrix A and sets the other entries as zero. For more details of compression operators, one can refer to [37]–[39].

The following proposition shows that both the Rank-K and Top-K compression operators satisfy Assumption 4.

Proposition 1. *For the Rank-K and Top-K compression operators, Assumption 4 holds with $\delta = \frac{K}{2d}$ and $\delta = \frac{K}{2d^2}$, respectively.*

Proof. See the full version [40]. □

Remark 1. *Different from the random compression operators used in first-order algorithms [41], [42], we use deterministic compression operators. This is because any realization not satisfying* (3) *may lead to a non-positive semidefinite Hessian approximation and thus leads to the failure of the proposed Newton's method.*

B. Algorithm Development

In this section, we propose a decentralized Newton's method with multi-step consensus and compression. In iteration k, node i first conducts one minimization step along a local approximated Newton direction d_i and then communicates the result with its neighbors for m rounds to compute

$$x_i^{k+1} = \sum_{j\in\mathcal{N}_i} (W^m)_{ij}\left(x_j^k - \alpha d_j^k\right). \tag{4}$$

Here, $\alpha > 0$ is a step size and $(W^m)_{ij}$ is the (i,j)-th entry of W^m. Such multi-step consensus costs m rounds of communication. As we will show in the next section, multi-step consensus balances between computation and communication and is vital to get a provably fast convergence rate.

To update the local direction, we use the DAC technique to obtain a gradient approximation and a Hessian approximation, which track the global gradient and global Hessian, respectively. The gradient approximation g_i^{k+1} on node i is computed by

$$g_i^{k+1} = \sum_{j\in\mathcal{N}_i} (W^m)_{ij}\left(g_j^k + \nabla f_j(x_j^{k+1}) - \nabla f_j(x_j^k)\right) \tag{5}$$

with initialization $g_i^0 = \nabla f_i(x_i^0)$. Similar to (4), we use multi-step consensus to make g_i^{k+1} a more accurate gradient approximation.

Algorithm 1: Decentralized Newton's method

Input: $\mathbf{x}^0, \mathbf{d}^0, g_i^0 = \nabla f_i(x_i^0), H_i^0 = \nabla^2 f_i(x_i^0)$, E_i^0, $\tilde{H}_i^0$, α, γ, m, M.

for $k = 0, 1, 2, \ldots$ **do**

$\mathbf{x}^{k+1} = \mathbf{W}^m(\mathbf{x}^k - \alpha\mathbf{d}^k)$

$\mathbf{g}^{k+1} = \mathbf{W}^m\left(\mathbf{g}^k + \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k)\right)$

Compression Procedure

$\tilde{\mathbf{H}}^{k+1} = \tilde{\mathbf{H}}^k + \mathcal{Q}(\mathbf{H}^k - \tilde{\mathbf{H}}^k)$

$\hat{\mathbf{H}}^k = \tilde{\mathbf{H}}^k + \mathcal{Q}(\mathbf{E}^k + \mathbf{H}^k - \tilde{\mathbf{H}}^k)$

$\mathbf{E}^{k+1} = \mathbf{E}^k + \mathbf{H}^k - \tilde{\mathbf{H}}^k - \mathcal{Q}(\mathbf{E}^k + \mathbf{H}^k - \tilde{\mathbf{H}}^k)$

$\mathbf{H}^{k+1} = \mathbf{H}^k - \gamma(I_{nd} - \mathbf{W})\hat{\mathbf{H}}^k + \nabla^2 f(\mathbf{x}^{k+1}) - \nabla^2 f(\mathbf{x}^k)$

$\mathbf{d}^{k+1} \approx \left(\mathrm{diag}\{H_i^{k+1}\} + MI_{nd}\right)^{-1}\mathbf{g}^{k+1}$

end for

To obtain the Hessian approximation, we also use DAC to mix the second-order curvature information over the network but keep in mind that communicating the entire local Hessian approximation leads to a high communication cost. Thus, we design a compression procedure with error compensation to estimate the global Hessian in a communication-efficient way. In other words, each node is able to obtain more accurate global curvature information by exchanging the compressed

local Hessian approximation with its neighbors, without incurring a high communication cost. The Hessian approximation H_i^{k+1} on node i is given by

$$\begin{aligned}
&\tilde{H}_i^{k+1} = \tilde{H}_i^k + \mathcal{Q}(H_i^k - \tilde{H}_i^k),\\
&\hat{H}_i^k = \tilde{H}_i^k + \mathcal{Q}(E_i^k + H_i^k - \tilde{H}_i^k),\\
&E_i^{k+1} = E_i^k + H_i^k - \tilde{H}_i^k - \mathcal{Q}(E_i^k + H_i^k - \tilde{H}_i^k),\\
&H_i^{k+1} = H_i^k - \gamma \sum_{j\in\mathcal{N}_i} w_{ij}(\hat{H}_i^k - \hat{H}_j^k) + \nabla^2 f_i(x_i^{k+1}) - \nabla^2 f_i(x_i^k)
\end{aligned} \tag{6}$$

with initialization $H_i^0 = \nabla^2 f_i(x_i^0)$, where $\gamma > 0$ is a parameter. Compared with DAC without compression, i.e., $H_i^{k+1} = H_i^k - \gamma \sum_{j\in\mathcal{N}_i} w_{ij}(H_i^k - H_j^k) + \nabla^2 f_i(x_i^{k+1}) - \nabla^2 f_i(x_i^k)$, the term $w_{ij}(H_i^k - H_j^k)$ is replaced by $w_{ij}(\hat{H}_i^k - \hat{H}_j^k)$, which can be constructed with compressed communication. There are two techniques to compensate for the compression error in the construction of $\hat{H}_i^k$: (i) We introduce $\tilde{H}_i^k$ as a counterpart of H_i^k and compress their difference $H_i^k - \tilde{H}_i^k$. (ii) We add E_i^k—the compression error in the $(k-1)$-st iteration—back into the difference $H_i^k - \tilde{H}_i^k$ in the k-th iteration for error feedback and compress $E_i^k + H_i^k - \tilde{H}_i^k$. Intuitively, there is no compression error when the algorithm converges so that $H_i^k \to \tilde{H}_i^k$ and $E_i^k \to 0$. This intuition will be verified by our analysis later (see Proposition 3). It is worth noting that we only use one round of communication per iteration to construct the Hessian approximation.

With the local gradient and Hessian approximations, we are ready to update the local direction. To avoid calculating the inverse of the local Hessian approximation, we utilize an early-terminated conjugate gradient (CG) method [43] to obtain a local direction d_i^{k+1} via

$$\left(H_i^{k+1} + MI_d\right) d_i^{k+1} \approx g_i^{k+1},$$

where $M > 0$ is a regularization parameter. The accuracy of the CG step will be given later (see Fact 1). The proposed algorithm can be written in a compact form as summarized in Algorithm 1. With a slight abuse of notation in the compact form, given an aggregated matrix $A = [A_1; \ldots; A_n] \in \mathbb{R}^{nd\times d}$, we use $\mathcal{Q}(\cdot)$ to denote the block-wise compression operator such that $\mathcal{Q}(A) = [\mathcal{Q}(A_1); \ldots; \mathcal{Q}(A_n)] \in \mathbb{R}^{nd\times d}$.

In Algorithm 1, computing $\mathbf{W}\hat{\mathbf{H}}^k$ requires communicating the uncompressed matrices $\hat{H}_1^k, \ldots, \hat{H}_n^k$, which is costly. As shown in [44], there is an equivalent but communication-efficient implementation of the compression procedure, summarized in Algorithm 2. The basic idea is to introduce an auxiliary variable $\tilde{\mathbf{H}}_w^k$ that is equal to $\mathbf{W}\tilde{\mathbf{H}}^k$ and use it to construct $\hat{\mathbf{H}}_w^k$ that is equal to $\mathbf{W}\hat{\mathbf{H}}^k$. Algorithm 2 is communication-efficient since the nodes only communicate the compressed variables $\mathbf{Q}^k$ and $\hat{\mathbf{Q}}^k$. For simplicity, we study Algorithm 1 in our convergence analysis.

Remark 2. *The work [27] proposes a Newton-Raphson method, which utilizes DAC to track the global gradient and global Hessian. However, the method requires communication of local Hessians and computation of matrix inverses. There is no analysis of the convergence rate in [27]. The work [39] proposes a decentralized primal-dual algorithm called LEAD, which compresses the local gradient information to save communication. Our compression procedure is similar to that of LEAD but is applied to compress the local Hessian information. Further efforts are required to analyze the compression procedure in our paper, including the uniform boundedness of the Hessian approximations (in Stage I) and the approximation errors with respect to the global Hessian (in Stage II). We will elaborate on these in Section III.*

Algorithm 2: Communication-efficient implementation

Input: $\mathbf{x}^0, \mathbf{d}^0, g_i^0 = \nabla f_i(x_i^0), H_i^0 = \nabla^2 f_i(x_i^0)$, E_i^0, $\tilde{H}_i^0$, $\tilde{\mathbf{H}}_w^0 = \mathbf{W}\tilde{\mathbf{H}}^0$, α, γ, m, M.

for $k = 0, 1, 2, \ldots$ **do**

$\mathbf{x}^{k+1} = \mathbf{W}^m(\mathbf{x}^k - \alpha\mathbf{d}^k)$

$\mathbf{g}^{k+1} = \mathbf{W}^m\left(\mathbf{g}^k + \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k)\right)$

Compression Procedure

$\mathbf{Q}^k = \mathcal{Q}(\mathbf{H}^k - \tilde{\mathbf{H}}^k)$

$\hat{\mathbf{Q}}^k = \mathcal{Q}(\mathbf{E}^k + \mathbf{H}^k - \tilde{\mathbf{H}}^k)$

$\tilde{\mathbf{H}}^{k+1} = \tilde{\mathbf{H}}^k + \mathbf{Q}^k$

$\tilde{\mathbf{H}}_w^{k+1} = \tilde{\mathbf{H}}_w^k + \mathbf{W}\mathbf{Q}^k$

$\hat{\mathbf{H}}^k = \tilde{\mathbf{H}}^k + \hat{\mathbf{Q}}^k$

$\hat{\mathbf{H}}_w^k = \tilde{\mathbf{H}}_w^k + \mathbf{W}\hat{\mathbf{Q}}^k$

$\mathbf{E}^{k+1} = \mathbf{E}^k + \mathbf{H}^k - \tilde{\mathbf{H}}^k - \hat{\mathbf{Q}}^k$

$\mathbf{H}^{k+1} = \mathbf{H}^k - \gamma(\hat{\mathbf{H}}^k - \hat{\mathbf{H}}_w^k) + \nabla^2 f(\mathbf{x}^{k+1}) - \nabla^2 f(\mathbf{x}^k)$

$\mathbf{d}^{k+1} \approx \left(\mathrm{diag}\{H_i^{k+1}\} + MI_{nd}\right)^{-1} \mathbf{g}^{k+1}$

end for

III. Convergence Analysis

In this section, we conduct a novel two-stage analysis of our proposed Algorithm 1 and establish its convergence rate. Our analysis reveals that Algorithm 1 is provably faster than the first-order algorithms. For the centralized Newton's method, to get a globally linear convergence rate with arbitrary initialization and a locally quadratic convergence rate, one often resorts to backtracking line search, which adaptively gives a small step size at the early stage and always gives a unit step size within a neighborhood of the optimal solution [45]. However, backtracking line search becomes expensive in the decentralized setting. Nevertheless, we can mimic the process of backtracking line search and split the convergence process into two stages: The algorithm uses a small step size in Stage I and converges linearly until the local copies are close enough to the optimal solution. Then, the algorithm enters Stage II and starts to use a unit step size; we will show a local faster-than-linear rate by taking advantage of the curvature information which is not exploited in Stage I.

Before starting the analysis, we specify the accuracy of the CG method with the following fact [43].

Fact 1. *With at most d iterations for each node, the CG step yields*

$$\left(\mathrm{diag}\{H_i^{k+1}\} + MI_{nd}\right)\mathbf{d}^{k+1} = \mathbf{g}^{k+1} + \mathbf{r}^{k+1} \tag{7}$$

with

$$\|\mathbf{r}^{k+1}\| \le c_k\|\mathbf{g}^{k+1}\|$$

for any $0 \leq c_k \leq 1$.

A. Stage I: Globally Linear Convergence

The convergence analysis of Stage I is inspired by the work of [33], where a general framework of stochastic decentralized quasi-Newton methods is proposed. A globally linear convergence rate is established under the assumption that the constructed Hessian inverse approximations are positive definite with bounded eigenvalues. Similar to [33], we define two constants M_1 and M_2 as

$$M_1 \triangleq \mu + M - L_2\sqrt{\frac{u_1^0}{n}} - \tilde{u}_2^0,\ M_2 \triangleq L_1 + M + L_2\sqrt{\frac{u_1^0}{n}} + \tilde{u}_2^0, \tag{8}$$

where u_1^0 is defined in (10) and $\tilde{u}_2^0$ is a constant given in (40). When choosing the parameter $M \geq \tilde{u}_2^0 + L_2\sqrt{\frac{u_1^0}{n}}$, we have $M_2 \geq M_1 > 0$. We establish the globally linear convergence in Stage I under the condition

$$M_1 I_d \preceq H_i^k + M I_d \preceq M_2 I_d,\ \forall i \in \{1, \ldots, n\}. \tag{9}$$

We will prove that the sequence $\{H_i^k\}_{k\geq 0}$ generated by Algorithm 1 satisfy condition (9) for all $k \geq 0$ (see Proposition 4).

1) Main Theorem for Stage I: In Stage I, we want to establish the globally linear convergence of Algorithm 1 with arbitrary initialization. To this end, it is sufficient to only use the uniform bound instead of the curvature information of Hessian approximations given in (9). This significantly simplifies our analysis of Stage I. To begin, let us define

$$\mathbf{q}_1^k \triangleq \begin{pmatrix} \|\mathbf{x}^k - \mathbf{W}^\infty \mathbf{x}^k\|^2 \\ \frac{1}{L_1^2}\|\mathbf{g}^k - \mathbf{W}^\infty \mathbf{g}^k\|^2 \\ \frac{n}{L_1}\left(F(\overline{\mathbf{x}}^k) - F(x^*)\right) \end{pmatrix}$$

and

$$u_1^k \triangleq \left(1, \frac{(1-\sigma^2)^2}{50}, 2\sigma^{m-1}\right)\mathbf{q}_1^k, \tag{10}$$

where $u_1^k = 0$ implies that $x_1^k = \cdots = x_n^k = x^*$ due to the strong convexity of F.

Theorem 1. *Under Assumptions 1–4, if the parameters satisfy*

$$M \geq L_2\sqrt{\frac{u_1^0}{n}} + \tilde{u}_2^0,\ \alpha \leq \min\left\{\frac{M_1^2(1-\sigma^2)^3}{100L_1M_2\sigma^{m-1}}, \frac{M_1^2}{200L_1M_2}\right\},$$
$$c_k \leq \frac{M_1}{4M_2\sqrt{2\kappa_F}},\ \gamma \leq \frac{\delta^2(1-\sigma)}{50}, \tag{11}$$

then for any $m \geq 1$ *and any* $\mathbf{x}^0$*, we have*

$$u_1^{k+1} \leq \left(1 - \frac{\mu\alpha}{2M_2}\right)u_1^k, \quad \forall k \geq 0. \tag{12}$$

Theorem 1 implies that if the parameter M is sufficiently large, the step sizes α and γ are sufficiently small, and the CG step is sufficiently accurate, then Algorithm 1 converges linearly with any number of communication rounds m and any initialization $\mathbf{x}^0$.

Remark 3. *By substituting* (11) *into* (12)*, we have that the total number of iterations for Algorithm 1 to get an* ϵ*-optimal solution is*

$$O\left(\kappa_F \max\left\{\frac{\sigma^{m-1}}{(1-\sigma^2)^3}, 2\right\}\log\frac{1}{\epsilon}\right),$$

where we use $\frac{M_2}{M_1} = O(1)$ *by setting* $M \gg L_1 + L_2\sqrt{\frac{u_1^0}{n}} + \tilde{u}_2^0$*. Note that this complexity holds for any* $m \geq 1$*. If we set* $m \geq \frac{\log 2(1-\sigma^2)^3}{\log\sigma} + 1$*, then the computational complexity becomes*

$$O\left(\kappa_F \log\frac{1}{\epsilon}\right),$$

which is independent of the graph. The computational complexity of Stage I is still κ_F*-dependent because we only use the uniform bounds on the Hessian approximations and have not yet employed the curvature information. Nevertheless, this rate is still favorable since the goal of Stage I is to guarantee globally linear convergence with arbitrary initialization. We will show a faster theoretical rate for Stage II.*

Remark 4. *Compared with the analysis of stochastic quasi-Newton methods in [33], we consider the deterministic case and need novel techniques to control the inexactness caused by the CG step. Besides, we get a better theoretical computational complexity than that of [33] by using DAC to mix local Hessian approximations.*

2) One-step Descent in Stage I: Given that condition (9) holds at a certain time step k_0, the following proposition establishes one-step descent from $u_1^{k_0}$ to $u_1^{k_0+1}$.

Proposition 2. *Under Assumptions 1–4, if* $c_k \leq 1$ *and* (9) *holds at a certain time step* k_0*, then we have*

$$\mathbf{q}_1^{k_0+1} \leq \mathbf{J}^{[1]}\mathbf{q}_1^{k_0} \tag{13}$$

with

$$\mathbf{J}^{[1]} \triangleq \begin{bmatrix} 1-0.49(1-\sigma^2) & 0.004(1-\sigma^2)^3 & \frac{64\sigma^{2m}\alpha^2L_1^2}{(1-\sigma^2)M_1^2} \\ \frac{33}{1-\sigma^2} & 1-0.49(1-\sigma^2) & \frac{64\sigma^{2m}\alpha^2L_1^2}{(1-\sigma^2)M_1^2} \\ \frac{(1-\sigma^2)^3}{20\sigma^{m-1}} & \frac{(1-\sigma^2)^3}{80\sigma^{m-1}} & 1-\frac{\mu\alpha}{M_2} \end{bmatrix}.$$

Further, if the parameters M*,* α*,* c_k*, and* γ *satisfy* (11)*, then* (13) *implies that*

$$u_1^{k_0+1} \leq \left(1 - \frac{\mu\alpha}{2M_2}\right)u_1^{k_0}. \tag{14}$$

Proof. See Appendix VI-B. □

3) Proof of Main Theorem for Stage I: Thanks to Proposition 2, to prove Theorem 1, we only need to show that (9) holds for all $k \geq 0$. Observing that (9) gives bounds on the Hessian approximations, we need to take the specific compression procedure into consideration and give the convergence rate of the Hessian tracking error $\|\mathbf{H}^{k+1} - \mathbf{W}^\infty\mathbf{H}^{k+1}\|_F$. To do this, we establish the following proposition to bound the compression error $\|\mathbf{E}^{k+1}\|_F$, the Hessian approximation difference $\|\mathbf{H}^{k+1} - \tilde{\mathbf{H}}^{k+1}\|_F$, and the Hessian tracking error $\|\mathbf{H}^{k+1} - \mathbf{W}^\infty\mathbf{H}^{k+1}\|_F$. Let us define

$$\mathbf{q}_2^k \triangleq \begin{pmatrix} \|\mathbf{E}^k\|_F \\ \|\mathbf{H}^k - \tilde{\mathbf{H}}^k\|_F \\ \|\mathbf{H}^k - \mathbf{W}^\infty\mathbf{H}^k\|_F \end{pmatrix}$$

and

$$u_2^k \triangleq \left(\frac{\delta(1-\sigma)}{8(1-\delta)}, \frac{1-\sigma}{4}, 1\right)\mathbf{q}_2^k.$$

Here, $u_2^k = 0$ implies that $\mathbf{E}^k = 0$, $\tilde{\mathbf{H}}^k = \mathbf{H}^k$, and $H_1^k = \cdots = H_n^k = \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(x_i^k)$, where we use $\overline{\mathbf{H}}^{k+1} = \overline{\nabla^2 f}(\mathbf{x}^{k+1})$ for all $k \geq 0$.

Proposition 3. *Under Assumptions 1, 2, and 4, if $\gamma \leq 1$, then for all $k \geq 0$, we have*

$$\mathbf{q}_2^{k+1} \leq \mathbf{J}^{[2]}\mathbf{q}_2^k + L_2\|\mathbf{x}^{k+1} - \mathbf{x}^k\|[0;1;1] \tag{15}$$

with

$$\mathbf{J}^{[2]} \triangleq \begin{bmatrix} 1-\delta & 1-\delta & 0 \\ 4\gamma & (1-\delta+2\gamma(1-\delta)) & 2\gamma \\ 4\gamma & 2\gamma(1-\delta) & 1-\gamma(1-\sigma) \end{bmatrix}.$$

Further, under Assumption 3, if the parameters M, α, c_k, and γ satisfy (11) *and* (9) *holds at a certain time step k_0, then* (15) *implies that*

$$u_2^{k_0+1} \leq \left(1 - \frac{\gamma}{2}(1-\sigma)\right)u_2^{k_0} + \frac{15L_2}{4}\sqrt{\sigma^{-(m-1)}u_1^{k_0}}. \tag{16}$$

Proof. See Appendix VI-C. □

Observing that Propositions 2 and 3 hold for a certain k_0, we show that (9) holds for all $k \geq 0$ via mathematical induction.

Proposition 4. *Under the setting of Theorem 1, considering the sequence $\{H_i^k\}_{k\geq 0}$ for any $i \in \{1,\ldots,n\}$ generated by Algorithm 1, we have that condition* (9) *holds for all $k \geq 0$.*

Proof. See Appendix VI-D. □

Thus, combining Propositions 2 and 4 directly gives (12). This completes the proof of Theorem 1.

B. Stage II: Faster Local Convergence

After Stage I, all the local copies $x_1^k, \ldots, x_n^k$ are close enough to x^* according to Theorem 1 and all the local Hessian approximations $H_1^k, \ldots, H_n^k$ are almost consensual according to Proposition 3, as long as k is sufficiently large. We will specify the number of iterations needed by Stage I later (see (18)). In Stage I, we only use the uniform boundedness of the Hessian approximations and do not take advantage of the curvature information adequately. However, in Stage II, to get a locally faster rate, we need to bound the error between each local Hessian approximation H_i^k and the global Hessian $\nabla^2 F(\overline{\mathbf{x}}^k)$. After this, we further bound the error between local directions d_i^k and the global Newton direction $(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)$ (see (42)). In this way, we utilize the locally quadratic convergence rate of the centralized Newton's method to bound $\|\overline{\mathbf{x}}^k - x^*\|$ (see Corollary 4). The analysis is novel compared with those of existing first-order and second-order methods and is vital to relate the decentralized Newton's method with the centralized one.

Let the proposed algorithm enter Stage II after K iterations with

$$K \geq \frac{\frac{m}{2}\log\sigma - \log\frac{41\kappa_F}{\mu\sqrt{n}}\left(\tilde{u}_2^0 + \frac{52L_2\kappa_F\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}{\log\phi}, \tag{18}$$

where ϕ is defined in (39). Let $M_1 \triangleq \frac{40\mu}{41}$ and $M_2 \triangleq L_1 + \frac{\mu}{41}$ in Stage II, which are different from those in Stage I but we use the same notation for simplicity. We establish the faster local rate in Stage II under the condition

$$M_1 I_d \preceq H_i^{k+1} \preceq M_2 I_d, \quad \forall\, i \in \{1,\ldots,n\} \tag{19}$$

for all $k \geq K$. Proposition 6 below shows that the sequence $\{H_i^k\}_{k\geq 0}$ for any $i \in \{1,\ldots,n\}$ generated by Algorithm 1 satisfies (19).

1) Main Theorem for Stage II: Define $\mathbf{q}_3^k$ and $\mathbf{J}^{[3]}$ as in (17) and

$$u_3^k \triangleq \left(1, \sigma^{-\frac{m}{4}}, 0.5\sigma^{-\frac{3m}{4}}\right)\mathbf{q}_3^k.$$

We establish a faster local rate for Stage II.

Theorem 2. *Under Assumptions 1–4, if the parameters satisfy*

$$\alpha = 1,\ M = 0,\ m > \frac{4\log(4\kappa_F)}{-\log\sigma},$$
$$c_k \leq \frac{M_1\sigma^{\frac{m}{2}}}{40\mu\kappa_F},\ \gamma \leq \frac{\delta^2(1-\sigma)}{50}, \tag{20}$$

then for all $k \geq K$, we have

$$u_3^{k+1} \leq \sigma^{\frac{m}{2}}u_3^k.$$

Theorem 2 implies that we get a κ_F-independent linear rate when the number of consensus inner steps m is constant in all iterations. The following corollary extends Theorem 2 to the case where the number of consensus inner steps varies with the number of iterations.

Corollary 1. *Under Assumptions 1–4, if the parameters satisfy*

$$\alpha = 1,\ M = 0,\ m > \frac{4\log(4\kappa_F)}{-\log\sigma},$$
$$c_k \leq \frac{M_1\sigma^{\frac{m_k}{2}}}{40\mu\kappa_F},\ \gamma \leq \frac{\delta^2(1-\sigma)}{50},$$

where

$$m_k = m + \left\lfloor \frac{2(k-K)\log\phi}{\log\sigma} \right\rfloor \tag{21}$$

is the number of consensus inner steps in iteration k, then for all $k \geq K$, we have

$$u_3^{k+1} \leq \sigma^{\frac{m_k}{2}}u_3^k.$$

Besides generalizing Theorem 2, Corollary 1 reveals the interesting theoretical phenomenon that if we choose $m_k \nearrow \infty$ as $k \nearrow \infty$, then we can achieve an asymptotically super-linear rate in Stage II.

Remark 5. *In Stage II, we artificially set a unit step size α to establish a faster local rate. This is done by taking advantage of the curvature information. Note that existing techniques for adaptively choosing step sizes, such as backtracking line search, require evaluations of the entire objective function for many times per iteration. In our numerical experiments, we show that geometrically increasing the step size to unit works well. Furthermore, here are our selection strategies for other parameters in Algorithm 2. We set γ to a small constant. We can simply set $M = 0$, because we use Hessian tracking to*

$$\mathbf{q}_3^k \triangleq \begin{pmatrix} \|\mathbf{x}^k - \mathbf{W}^\infty \mathbf{x}^k\| \\ \frac{1}{L_1}\|\mathbf{g}^k - \mathbf{W}^\infty \mathbf{g}^k\| \\ \sqrt{n}\|\overline{\mathbf{x}}^k - x^*\| \end{pmatrix}, \mathbf{J}^{[3]} \triangleq \begin{bmatrix} \sigma^m\left(1+\alpha\kappa_F\varrho^{\tilde{k}_0}\right) & \sigma^m\alpha(1+\varrho^{\tilde{k}_0})\kappa_F & 2\sigma^m\alpha\varrho^{\tilde{k}_0}\kappa_F \\ \sigma^m\left(2+\alpha\kappa_F+2\alpha\kappa_F\varrho^{\tilde{k}_0}\right) & \sigma^m\left(1+\alpha\kappa_F(\sigma^m+2\varrho^{\tilde{k}_0})\right) & \sigma^m\alpha\left(1+2\kappa_F\varrho^{\tilde{k}_0}+\vartheta^{\tilde{k}_0}\right) \\ \alpha\kappa_F\left(1+\varrho^{\tilde{k}_0}\right) & \alpha\kappa_F\varrho^{\tilde{k}_0} & 1-\alpha+\alpha\vartheta^{\tilde{k}_0}+\alpha\kappa_F\varrho^{\tilde{k}_0} \end{bmatrix}, \quad (17)$$

where $\kappa_F = \frac{L_1}{\mu}$, $\vartheta^k \triangleq \frac{L_2}{2\mu}\|\overline{\mathbf{x}}^k - x^*\|$, and $\varrho^k \triangleq \frac{1}{M_1}\left(\frac{L_2}{\sqrt{n}}\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\| + \frac{1}{\sqrt{n}}\|\mathbf{H}^k - \mathbf{W}^\infty\mathbf{H}^k\|_F + c_k\mu\right)$ for all $k \geq 0$.

make H_i^k a good approximation of the global Hessian. We use the CG method to avoid computing the inverse of H_i^k. We fix the parameter c_k, which represents the accuracy of the CG step, to a small constant. Since the CG step just solves a linear system approximately locally, it can be completed quickly. We set m to a fixed constant. When the geometrically increasing step size α reaches 1, the proposed algorithm automatically enters Stage II, in which we need to adjust m to an appropriate value to achieve a better balance between computation and communication. As an aside, we can use an increasing m_k (for example, $m_k = k$) to verify the asymptotically super-linear rate as guaranteed by Corollary 1, though this is more of theoretical interest.

2) One-step Descent in Stage II: To prove Theorem 2, given that condition (19) holds for a certain time step $\tilde{k}_0$ with $\tilde{k}_0 \geq K$, we establish one-step descent from $u_2^{\tilde{k}_0}$ to $u_2^{\tilde{k}_0+1}$.

Proposition 5. *Under Assumptions 1–3, if $M = 0$ and (19) holds for a certain $\tilde{k}_0$ with $\tilde{k}_0 \geq K$, then we have*

$$\mathbf{q}_3^{\tilde{k}_0+1} \leq \mathbf{J}^{[3]}\mathbf{q}_3^{\tilde{k}_0}. \quad (22)$$

Further, if

$$\kappa_F\varrho^{\tilde{k}_0} + \vartheta^{\tilde{k}_0} \leq \frac{1}{20}\sigma^{\frac{m}{2}} \quad (23)$$

and the parameters α, m, c_k, and γ satisfy (20), then (22) implies that

$$u_3^{\tilde{k}_0+1} \leq \sigma^{\frac{m}{2}}u_3^{\tilde{k}_0}. \quad (24)$$

Proof. See Appendix VI-E. □

3) Proof of Theorem 2: According to Proposition 5, to prove Theorem 2, we only need to show (19) and (23) hold for all $k \geq K$. This is done in Proposition 6.

Proposition 6. *Under the setting of Theorem 2, (19) and (23) hold for any $k \geq K$.*

Proof. See Appendix VI-F. □

By combining Propositions 5 and 6, we complete the proof of Theorem 2.

4) Proof of Corollary 1: This is a direct extension of Propositions 5 and 6; see Appendix VI-G.

IV. Numerical Experiments

In the numerical experiments, we consider quadratic programming and logistic regression problems over a network. The network is randomly generated with n nodes connected by $\frac{\tau n(n-1)}{2}$ edges, where $\tau \in (0, 1]$ is the connectivity ratio.

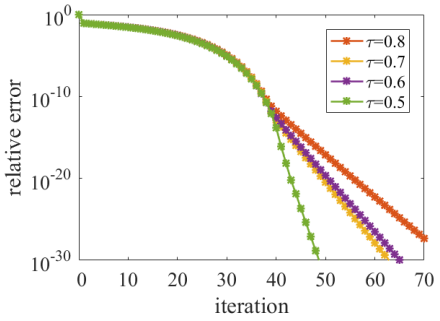

Fig. 1: Robustness to graph connectivity in Stage I

We pre-compute the optimal solution x^* with a centralized Newton's method. The performance metric is the relative error, defined as $\frac{1}{n}\|\mathbf{x}^k - \mathbf{x}^*\|^2/\|\mathbf{x}^0 - \mathbf{x}^*\|^2$. We compare the proposed method with the first-order method ABC [17], the accelerated first-order method $\mathcal{AB}m$ [15], the multi-step consensus accelerated first-order method Mudag [20], the second-order methods SONATA [30] and DiRegINA [25]. We use hand-optimized step sizes for all the algorithms. The experiments are done on a laptop with 1.80GHz Intel(R) Core(TM) i7 CPU, 16.0 GB RAM, and Windows 10 operating system.

A. Quadratic Programming

We conduct two sets of numerical experiments to show that the convergence rate of the proposed Newton's method is independent of the graph in Stage I when m is sufficiently large (as stated in Remark 3) and is independent of κ_F in Stage II (as stated in Theorem 2). Let us consider solving a quadratic programming problem over a network, i.e.,

$$x^* = \arg\min_{x\in\mathbb{R}^d} \sum_{i=1}^n \left(\frac{1}{2}x^T Q_i x + p_i^T x\right).$$

Each node i has private data $Q_i \in \mathbb{R}^{d\times d} \succ 0$ and $p_i \in \mathbb{R}^d$, whose elements are generated according to the standard Gaussian distribution. In both experiments, we use Rank-3 compression, set $n = 10$, $d = 30$, $M = 0$, $\gamma = 0.02$, and use the increasing step sizes $\alpha_k = \min\{1, 0.02 \times 1.1^k\}$.

To show that the convergence rate of the proposed Newton's method is independent of the graph when m is sufficiently large in Stage I, we run the proposed algorithm on random graphs with $\tau = 0.8, 0.7, 0.6, 0.5$, respectively. For these graphs, we have $\sigma = 0.570, 0.623, 0.639, 0.791$, respectively. According to Remark 3, to get a σ-independent convergence rate at Stage I, we should have $m \geq \frac{\log 2(1-\sigma^2)^3}{\log\sigma} + 1$. Thus, we set $m = 2, 3, 3, 11$, respectively. Also, we set $\kappa_F = 100$.

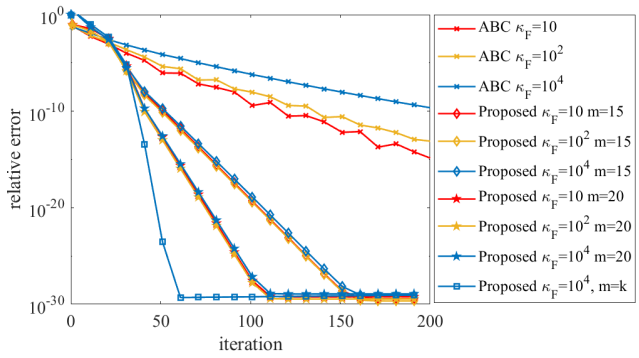

Fig. 2: Robustness to κ_F in Stage II

Fig. 1 shows that the convergence process of the proposed algorithm has two stages. The turning point is at round the 41-st iteration and the corresponding step size is $\alpha_k \approx 1$. In Stage I, the proposed algorithm has the same performance on all the tested graphs, which suggests that the convergence rate is independent of the graph when m is sufficiently large. This result is in line with our theoretical findings presented in Remark 3.

To show the independence of the condition number κ_F in Stage II (see Theorem 2), we generate matrices $Q_1, \ldots, Q_n$ under different condition numbers $\kappa_F = 10,\ 10^2,\ 10^4$. We set $\tau = 0.2$. We compare the proposed Newton's method with ABC. For the proposed Newton's method, to further show that its convergence rate is connected with the number of multi-step consensus inner-loops, we set $m = 15$ and $m = 20$. To show the asymptotic rate with $m \to \infty$, we consider $\kappa_F = 10^4$ and $m = k$, where k is the index of iteration. In ABC, we set the step sizes $\alpha = 0.4,\ 0.05,\ 0.0008$ for the condition numbers $\kappa_F = 10,\ 10^2,\ 10^4$, respectively.

Fig. 2 demonstrates the relative error versus the number of iterations. For the first-order method ABC, the convergence slows down with an increasing κ_F. On the other hand, for the proposed algorithm, similarly to the results displayed in Fig. 1, the convergence process has two stages. For each fixed m, our proposed Newton's method converges at the same rate under different κ_F. Also, the convergence rate is faster with a larger m. When we increase the number of multi-step consensus inner-loops to $m = k$, the convergence rate becomes much faster. These experiment results corroborate the theoretical findings in Theorem 2.

B. Logistic Regression

We conduct two sets of numerical experiments to compare our proposed method with the first-order methods ABC [17], $\mathcal{AB}m$ [15], and Mudag [20] with Top-K compressors, as well as the second-order methods SONATA [30] and DiRegINA [25] with Rank-K compressors. We solve a logistic regression problem

$$x^* = \underset{x\in\mathbb{R}^d}{\operatorname{argmin}}\ \frac{\rho}{2}\|x\|^2 + \sum_{i=1}^{n}\sum_{j=1}^{m_i} \ln\left(1+\exp\left(-\left(\mathbf{o}_{ij}^T x\right)\mathbf{p}_{ij}\right)\right),$$

in which each node i privately owns m_i training samples $(\mathbf{o}_{ij}, \mathbf{p}_{ij}) \in \mathbb{R}^d \times \{-1, +1\}$, $j = 1, \ldots, m_i$. The regularization term $\frac{\rho}{2}\|x\|^2$ parameterized by $\rho > 0$ is to avoid overfitting.

1) **Comparison with first-order methods***:* In the first set of numerical experiments, the elements of $\mathbf{o}_{ij}$ are randomly generated following the standard Gaussian distribution and those of $\mathbf{p}_{ij}$ are generated following the uniform distribution on $\{-1, 1\}$. Thus, $f_1, \ldots, f_n$ are similar. We use the Top-K compression operator, where node i transmits $K = 20$ entries of the matrix with the largest absolute values and their indexes. There are $n = 30$ nodes with connectivity ratio $\tau = 0.2$. Each node has 100 samples, i.e. $m_i = 100, \forall i$. The dimension is $d = 20$ and the regularization parameter is $\rho = 0.001$.

The step size of ABC, $\mathcal{AB}m$, and Mudag are best tuned. For the proposed algorithm, we use the increasing step sizes $\alpha_k = \min\left\{1, 0.2 \times 1.1^k\right\}$ and set $\gamma = 0.03$. We set $M = 0$ and the proposed algorithm still works well. We speculate that this is because we use the CG step to avoid computing the inverse of $H_i^k + MI_d$ and H_i^k becomes closer and closer to $\overline{\mathbf{H}}^k$ as the iteration proceeds.

Fig. 3 illustrates the relative error versus the number of iterations, the number of bits for communication, and the running time. We run the proposed algorithm with different m. $\mathcal{AB}m$ is better than ABC due to the introduction of acceleration. Although Mudag outperforms $\mathcal{AB}m$ in terms of the number of iterations by further introducing multi-step consensus, $\mathcal{AB}m$ is advantageous over Mudag in terms of the number of transmitted bits and running time. When $m = 1$, the proposed Newton's method is better than the others in terms of the number of iterations but inferior in terms of the number of transmitted bits and running time, due the use of second-order information. When $m \geq 7$, it outperforms the others in terms of all three metrics. The reason is that the advantage of multi-step consensus in reducing the number of iterations outweighs its disadvantage in incurring more transmitted bits and running time. In addition, the properly designed compression procedure guarantees that the number of transmitted bits is limited.

2) **Comparison with second-order methods without data similarity***:* We compare the proposed algorithm with the second-order methods SONATA and DiRegINA. It is worth noting that SONATA and DiRegINA are proved to have faster convergence rates than first-order methods under the assumption of data similarity, meaning that the local Hessians are similar so that each node can use its local Hessian as a substitute of the global one [25], [30]. However, in our algorithm, we do not require such a data similarity assumption. To generate dissimilar data, the elements of $\mathbf{o}_{ij}$ are drawn from the Gaussian distribution with mean 0 and variance i, where $i \in [n]$ and $j \in [m_i]$. The other parameters are the same as those in Fig. 3. We use the Rank-K compression operator, where each node performs a singular value decomposition of the matrix and transmits the largest $K = 3$ singular values as well as the corresponding singular vectors. For the proposed algorithm, we use the increasing step sizes $\alpha_k = \min\left\{1, 0.2 \times 1.1^k\right\}$. We set $\gamma = 0.08$. For SONATA, we use the second-order approximation of the local objective $f_i(x) \approx f_i(x_i^k) + \langle g_i^k, x - x_i^k\rangle + 1/2(x - x_i^k)^T(\nabla^2 f_i(x_i^k) + \epsilon I_d)(x - x_i^k)$,

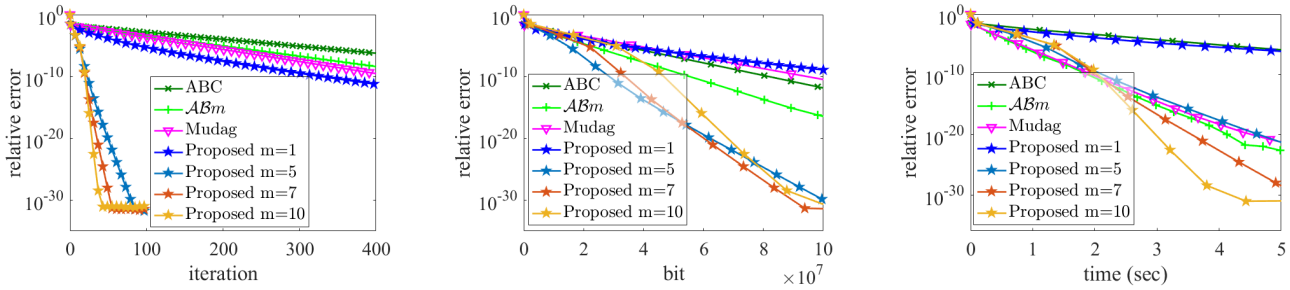

Fig. 3: Comparison with first-order methods with Top-K compressor

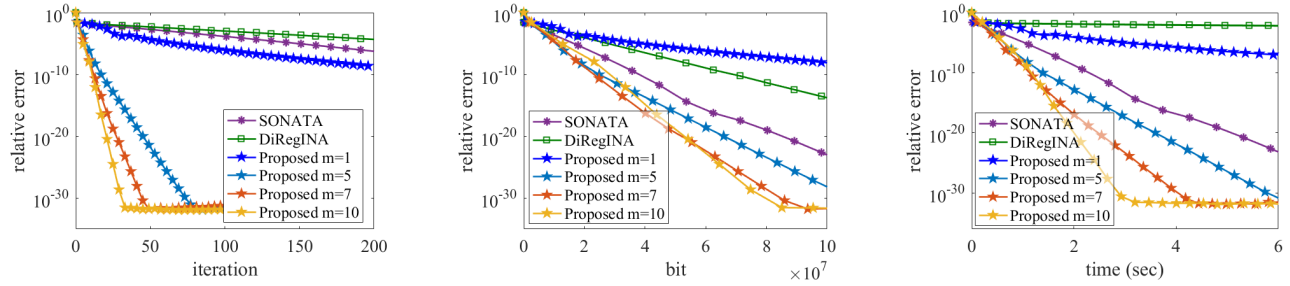

Fig. 4: Comparison with the second-order methods with Rank-K compressor

where g_i^k is the gradient approximation of node i at the k-th iteration. Here, we set $\epsilon = 0.8$. For DiRegINA, we use the second-order approximation of the local objective with cubic regularization $f_i(x) \approx f_i(x_i^k) + \langle g_i^k, x - x_i^k\rangle + 1/2(x - x_i^k)^T(\nabla^2 f_i(x_i^k) + \epsilon I_d)(x - x_i^k) + \zeta/6\|x - x_i^k\|^3$ and solve it using the code provided by [46]. We set $\epsilon = 1$ and $\zeta = 0.9$.

Fig. 4 shows the relative error versus the number of iterations, the number of transmitted bits, and the running time. When $m \geq 5$, the proposed Newton's method outperforms SONATA in terms of the number of iterations because we use DAC to track the global Hessian. Due to the compression procedure and the CG step, our method also has the best performance in terms of the number of transmitted bits and running time. We observe that SONATA is better than DiRegINA in terms of the number of iterations and running time. Without data similarity, the local Hessian is likely not a good substitute for the global Hessian. In this case, using cubic regularization does not improve the convergence performance. In addition, DiRegINA requires an iterative algorithm in the inner loop to solve the cubic regularized local subproblem, which is time-consuming.

V. Conclusions

This paper considers a finite-sum minimization problem over a decentralized network. We propose a communication-efficient decentralized Newton's method for solving it, which has provably faster convergence than first-order algorithms. Multi-step consensus that balances between computation and communication is used for communicating local copies of the decision variable and gradient approximations. We also use compression with error compensation for transmitting the local Hessian approximations, which utilizes the global second-order information while avoiding high communication cost. We present a novel convergence analysis and obtain a theoretically faster convergence rate than those of first-order algorithms. One future direction is to develop stochastic second-order algorithms with provably κ_F-independent super-linear convergence rate, considering the case when computing the local full gradient and Hessian is not affordable on each node. Another interesting direction is to develop decentralized second-order algorithms to solve nonconvex and nonsmooth problems that arise in various machine learning and signal processing applications.

VI. Appendix

A. Preliminary

This section gives some preliminaries that are useful in the ensuing convergence analysis. The following lemma bounds the consensus errors of the iterate $\mathbf{x}^k$ and the gradient approximation $\mathbf{g}^k$.

Lemma 1. *Under Assumptions 1 and 2, for all $k \geq 0$, we have*

$$\|\mathbf{x}^{k+1} - \mathbf{W}^\infty \mathbf{x}^{k+1}\| \leq \sigma^m(\|\mathbf{x}^k - \mathbf{W}^\infty \mathbf{x}^k\| + \alpha\|\mathbf{d}^k - \mathbf{W}^\infty \mathbf{d}^k\|) \tag{25}$$

and

$$\|\mathbf{g}^{k+1} - \mathbf{W}^\infty \mathbf{g}^{k+1}\| \leq \sigma^m(\|\mathbf{g}^k - \mathbf{W}^\infty \mathbf{g}^k\| + L_1\|\mathbf{x}^{k+1} - \mathbf{x}^k\|). \tag{26}$$

Proof. See Supplementary I. □

The following lemma bounds the norm of the gradient approximation $\mathbf{g}^k$ and the difference between two successive iterates.

Lemma 2. *Under Assumptions 1 and 2, if $c_k \leq 1$ and condition* (9) *holds for a certain k_0, then we have*

$$\|\mathbf{g}^k\| \leq \|\mathbf{g}^k - \mathbf{W}^\infty \mathbf{g}^k\| + L_1 \|\mathbf{x}^k - \mathbf{W}^\infty \mathbf{x}^k\| + \sqrt{n} \|\nabla F(\overline{\mathbf{x}}^k)\| \quad (27)$$

for all $k \geq 0$ and

$$\begin{aligned}\|\mathbf{x}^{k_0+1} - \mathbf{x}^{k_0}\| \leq & \left(2 + \frac{2\alpha L_1}{M_1}\right) \|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}^{k_0}\| \quad (28)\\ & + \frac{2\alpha}{M_1} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\| + \frac{2\alpha\sqrt{n}}{M_1} \|\nabla F(\overline{\mathbf{x}}^{k_0})\|.\end{aligned}$$

Proof. See Supplementary II. □

B. Proof of Proposition 2

Proof. First, we prove (13) with three lemmas. We will bound the consensus error $\|\mathbf{x}^{k_0+1} - \mathbf{W}^\infty \mathbf{x}^{k_0+1}\|^2$, the gradient tracking error $\frac{1}{L_1^2}\|\mathbf{g}^{k_0+1} - \mathbf{W}^\infty \mathbf{g}^{k_0+1}\|^2$, and the network optimality gap $\frac{n}{L_1}(F(\overline{\mathbf{x}}^{k_0+1}) - F(x^*))$ in Lemmas 3, 4, and 5 respectively.

Lemma 3. *Under Assumptions 1 and 2, if $c_k \leq 1$, condition* (9) *holds for a certain k_0, and α satisfy* (11)*, then we have*

$$\begin{aligned}&\|\mathbf{x}^{k_0+1} - \mathbf{W}^\infty \mathbf{x}^{k_0+1}\|^2\\ \leq & \mathbf{J}_{11}^{[1]} \|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}^{k_0}\|^2 + \mathbf{J}_{12}^{[1]} \cdot \frac{1}{L_1^2} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\|^2 \quad (29)\\ & + \mathbf{J}_{13}^{[1]} \cdot \frac{n}{L_1} \left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right).\end{aligned}$$

Proof. See Supplementary III. □

Lemma 4. *Under the setting of Lemma 3, we have*

$$\begin{aligned}&\frac{1}{L_1^2}\|\mathbf{g}^{k_0+1} - \mathbf{W}^\infty \mathbf{g}^{k_0+1}\|^2\\ \leq & \mathbf{J}_{21}^{[1]} \|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}^{k_0}\|^2 + \mathbf{J}_{22}^{[1]} \cdot \frac{1}{L_1^2} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\|^2 \quad (30)\\ & + \mathbf{J}_{23}^{[1]} \cdot \frac{n}{L_1} \left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right).\end{aligned}$$

Proof. See Supplementary IV. □

Lemma 5. *Under the setting of Lemma 3, we have*

$$\begin{aligned}&\frac{n}{L_1} \left(F(\overline{\mathbf{x}}^{k_0+1}) - F(x^*)\right)\\ \leq & \mathbf{J}_{31}^{[1]} \|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}_{k_0}\|^2 + \mathbf{J}_{32}^{[1]} \cdot \frac{1}{L_1^2} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\|^2 \quad (31)\\ & + \mathbf{J}_{33}^{[1]} \cdot \frac{n}{L_1} \left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right).\end{aligned}$$

Proof. See Supplementary V. □

By combining Lemmas 3–5, we get (13).

To prove (14) from (13), we substitute the parameters satisfying (11) and do algebraic manipulations. Please see the full version [40] for details. This completes the proof. □

C. Proof of Proposition 3

Proof. The proof of (15) is decomposed into three lemmas. We are going to bound the compression error $\|\mathbf{E}^{k+1}\|_F$, the difference $\|\mathbf{H}^{k+1} - \tilde{\mathbf{H}}^{k+1}\|_F$, and the Hessian tracking error $\|\mathbf{H}^{k+1} - \mathbf{W}^\infty \mathbf{H}^{k+1}\|_F$ in Lemmas 6, 7, and 8, respectively.

Lemma 6. *Under Assumption 4, for all $k \geq 0$, we have*

$$\|\mathbf{E}^{k+1}\|_F \leq (1-\delta)\|\mathbf{E}^k\|_F + (1-\delta)\|\mathbf{H}^k - \tilde{\mathbf{H}}^k\|_F.$$

Proof. See Supplementary VI. □

Lemma 7. *Under Assumptions 1, 2, and 4, for all $k \geq 0$, we have*

$$\begin{aligned}&\|\mathbf{H}^{k+1} - \tilde{\mathbf{H}}^{k+1}\|_F\\ \leq & (1 - \delta + 2\gamma(1-\delta)) \|\mathbf{H}^k - \tilde{\mathbf{H}}^k\|_F + 4\gamma \|\mathbf{E}^k\|_F \quad (32)\\ & + 2\gamma \|\mathbf{H}^k - \mathbf{W}^\infty \mathbf{H}^k\|_F + L_2 \|\mathbf{x}^{k+1} - \mathbf{x}^k\|.\end{aligned}$$

Proof. See Supplementary VII. □

Lemma 8. *Under Assumptions 1, 2, and 4, if $\gamma < 1$, then for all k, we have*

$$\begin{aligned}&\|\mathbf{H}^{k+1} - \mathbf{W}^\infty \mathbf{H}^{k+1}\|_F\\ \leq & (1 - \gamma(1-\sigma)) \|\mathbf{H}^k - \mathbf{W}^\infty \mathbf{H}^k\|_F + 4\gamma \|\mathbf{E}^k\|_F \quad (33)\\ & + 2\gamma(1-\delta) \|\mathbf{H}^k - \tilde{\mathbf{H}}^k\|_F + L_2 \|\mathbf{x}^{k+1} - \mathbf{x}^k\|.\end{aligned}$$

Proof. See Supplementary VIII. □

Combining Lemmas 6–8 directly gives (15).

Next, we prove (16) from (15). By choosing $\gamma \leq \frac{\delta^2(1-\sigma)}{50}$, it is easy to show that

$$\left(\frac{\delta(1-\sigma)}{8(1-\delta)}, \frac{1-\sigma}{4}, 1\right) \mathbf{J}^{[2]} \leq \left(1 - \frac{\gamma}{2}(1-\sigma)\right) \left(\frac{\delta(1-\sigma)}{8(1-\delta)}, \frac{1-\sigma}{4}, 1\right).$$

Thus, by multiplying $\left(\frac{\delta(1-\sigma)}{8(1-\delta)}, \frac{1-\sigma}{4}, 1\right)$ on both sides of (15), we get

$$u_2^{k+1} \leq \left(1 - \frac{\gamma}{2}(1-\sigma)\right) u_2^k + \frac{5L_2}{4} \|\mathbf{x}^{k+1} - \mathbf{x}^k\| \quad (34)$$

for all $k \geq 0$. Further, according to (28), we have

$$\begin{aligned}&\|\mathbf{x}^{k_0+1} - \mathbf{x}^{k_0}\|^2\\ \leq & 2\left(2 + \frac{2\alpha L_1}{M_1}\right)^2 \|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}^{k_0}\|^2 + \frac{16\alpha^2}{M_1^2} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\|^2\\ & + \frac{32\alpha^2 L_1}{M_1^2} n \left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right) \quad (35)\\ \leq & 9\Big(\|\mathbf{x}^{k_0} - \mathbf{W}^\infty \mathbf{x}^{k_0}\|^2 + \frac{(1-\sigma^2)^2}{50L_1^2} \|\mathbf{g}^{k_0} - \mathbf{W}^\infty \mathbf{g}^{k_0}\|^2\\ & + \frac{n}{L_1} \left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right)\Big) = \frac{9u_1^{k_0}}{\sigma^{m-1}},\end{aligned}$$

where we substitute α in (11) in the second inequality. By substituting (35) into (34), we get (16) and complete the proof. □

D. Proof of Proposition 4

Proof. We use mathematical induction to prove this proposition. First, it is easy to see that (9) holds for $k = 0$. Second, assume that (9) holds for all $0, 1, \ldots, k-1$. Then, Proposition 2 implies that

$$u_1^k \leq \left(1 - \frac{\mu\alpha}{2M_2}\right) u_1^{k-1} \leq \cdots \leq \left(1 - \frac{\mu\alpha}{2M_2}\right)^k u_1^0. \quad (36)$$

By substituting (36) into (16), we get

$$\begin{aligned} u_2^k \leq & \left(1 - \frac{\gamma}{2}(1-\sigma)\right) u_2^{k-1} \\ & + \frac{15L_2}{4}\sqrt{\sigma^{-(m-1)}u_1^0} \cdot \left(1 - \frac{\mu\alpha}{2M_2}\right)^{\frac{k-1}{2}}. \end{aligned} \quad (37)$$

By unrolling (37), we have

$$u_2^k \leq (u_2^0 - C)\left(1 - \frac{\gamma}{2}(1-\sigma)\right)^k + C\left(1 - \frac{\mu\alpha}{4M_2}\right)^k, \quad (38)$$

where $C \triangleq \frac{3.75L_2\sqrt{\sigma^{-(m-1)}u_1^0}}{\sqrt{1-\frac{\mu\alpha}{2M_2}} - \left(1-\frac{\gamma(1-\sigma)}{2}\right)}$. Let us define

$$\phi \triangleq \max\left\{1 - \frac{\gamma}{2}(1-\sigma), 1 - \frac{\mu\alpha}{4M_2}\right\}. \quad (39)$$

Then, (38) implies that

$$u_2^k \leq \phi^k \tilde{u}_2^0$$

with

$$\tilde{u}_2^0 \triangleq \max\left\{u_2^0 - C, C\right\}. \quad (40)$$

To complete the proof, it remains to show that (9) also holds at time step k. To do this, with (6), we have

$$\overline{\mathbf{H}}^k = \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(x_i^k).$$

A simple computation shows that

$$\begin{aligned} \|\overline{\mathbf{H}}^k - \nabla^2 F(\overline{\mathbf{x}}^k)\| \leq & \frac{1}{n}\sum_{i=1}^n \|\nabla^2 f_i(x_i^k) - \nabla^2 f_i(\overline{\mathbf{x}}^k)\| \\ \leq & \frac{L_2}{n}\sum_{i=1}^n \|x_i^k - \overline{\mathbf{x}}^k\| \leq \frac{L_2}{\sqrt{n}}\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\|. \end{aligned}$$

Then, we have

$$\begin{aligned} & \|H_i^k - \nabla^2 F(\overline{\mathbf{x}}^k)\| \\ \leq & \|H_i^k - \overline{\mathbf{H}}^k\| + \|\overline{\mathbf{H}}^k - \nabla^2 F(\overline{\mathbf{x}}^k)\| \\ \leq & \|\mathbf{H}^k - \mathbf{W}^\infty\mathbf{H}^k\| + \frac{L_2}{\sqrt{n}}\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\|. \end{aligned} \quad (41)$$

Based on the definitions of u_1^k and u_2^k, (41) implies that

$$\|H_i^k - \nabla^2 F(\overline{\mathbf{x}}^k)\| \leq u_2^k + L_2\sqrt{\frac{u_1^k}{n}},$$

where we use the fact that $\|\cdot\| \leq \|\cdot\|_F$. Since $\mu I_d \preceq \nabla^2 F(\overline{\mathbf{x}}^k) \preceq L_1 I_d$, we have

$$\left(\mu - L_2\sqrt{\frac{u_1^k}{n}} - u_2^k\right) I_d \preceq H_i^k \preceq \left(L_1 + L_2\sqrt{\frac{u_1^k}{n}} + u_2^k\right) I_d.$$

Since $u_1^k \leq u_1^0$, $u_2^k \leq \phi^k\tilde{u}_2^0 \leq \tilde{u}_2^0$, and $M \geq L_2\sqrt{\frac{u_1^0}{n}} + \tilde{u}_2^0$, based on the definitions of M_1 and M_2 given in (8), we have

$$M_1 I_d \preceq H_i^k + M I_d \preceq M_2 I_d.$$

Thus, we prove that (9) also holds at time step k and complete the proof. □

E. Proof of Proposition 5

Proof. First, we prove (22) in three steps. We are going to bound the consensus error $\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|$, the gradient tracking error $\|\mathbf{g}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\|$, and the network optimality gap $\|\overline{\mathbf{x}}^{\tilde{k}_0} - x^*\|$ in Step I, II, and III, respectively. Note that the first two terms have already been bounded in Theorem 1. The difference between Theorem 1 and Proposition 5 is that in Proposition 5 we dig deeper into the curvature information contained in the Hessian approximation $\mathbf{H}^k$ to bound the distance between $\mathbf{d}^k$ and the true Newton's direction, which gives tighter bounds for $\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|$ and $\|\mathbf{g}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\|$ than those given in Lemma 1.

Step I: To establish a tighter bound on the consensus error $\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|$, we need to bound $\|\mathbf{d}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{d}^{\tilde{k}_0}\|$ on the right-hand side of (25).

Lemma 9. *Under Assumptions 1 and 2, if condition* (19) *holds for a certain $\tilde{k}_0$ with $\tilde{k}_0 \geq K$, then we have*

$$\begin{aligned} & \|\overline{\mathbf{d}}^{\tilde{k}_0} - (\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0}))^{-1}\nabla F(\overline{\mathbf{x}}^{\tilde{k}_0})\| \\ \leq & \frac{L_1}{\mu\sqrt{n}}\left(1 + \varrho^{\tilde{k}_0}\right)\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\| \\ & + \frac{\varrho^{\tilde{k}_0}}{\mu\sqrt{n}}\left(\|\mathbf{g}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| + \sqrt{n}L_1\|\overline{\mathbf{x}}^{\tilde{k}_0} - x^*\|\right) \end{aligned} \quad (42)$$

and

$$\begin{aligned} \|\mathbf{d}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{d}^{\tilde{k}_0}\| \leq & \frac{1+\varrho^{\tilde{k}_0}}{\mu}\|\mathbf{g}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| \\ & + \frac{L_1\varrho^{\tilde{k}_0}}{\mu}\left(\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\| + \sqrt{n}\|\overline{\mathbf{x}}^{\tilde{k}_0} - x^*\|\right). \end{aligned} \quad (43)$$

Proof. See Supplementary IX. □

With Lemma 9, the following corollary gives a tighter bound on $\|\mathbf{x}^{k+1} - \mathbf{W}^\infty\mathbf{x}^{k+1}\|$.

Corollary 2. *Under the setting of Lemma 9, we have*

$$\begin{aligned} & \|\mathbf{x}^{\tilde{k}_0+1} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0+1}\| \\ \leq & \mathbf{J}_{11}^{[3]}\|\mathbf{x}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\| + \frac{\mathbf{J}_{12}^{[3]}}{L_1}\|\mathbf{g}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| + \mathbf{J}_{13}^{[3]}\sqrt{n}\|\overline{\mathbf{x}}^{\tilde{k}_0} - x^*\|. \end{aligned}$$

Proof. We substitute the tighter bound on $\|\mathbf{d}^{\tilde{k}_0} - \mathbf{W}^\infty\mathbf{d}^{\tilde{k}_0}\|$ given in (43) into (25) and complete the proof. □

Step II: To get a tighter bound on the gradient tracking error $\|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\|$, we need to bound $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|$ on the right hand of (26) by taking advantage of the curvature information.

Lemma 10. *Under Assumptions 1–3, if condition* (9) *holds for a certain $\tilde{k}_0$, then we have*

$$\begin{aligned}&\|\mathbf{x}^{\tilde{k}_0+1}-\mathbf{x}^{\tilde{k}_0}\|\\ \le&\left(2+\alpha\kappa_F+2\alpha\kappa_F\varrho^{\tilde{k}_0}\right)\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|\\ &+\alpha\kappa_F(\sigma^m+2\varrho^{\tilde{k}_0})\cdot\frac{1}{L_1}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| \qquad (44)\\ &+\alpha\left(1+2\kappa_F\varrho^{\tilde{k}_0}+\frac{L_2}{2\mu}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|\right)\sqrt{n}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|.\end{aligned}$$

Proof. See Supplementary X. □

With the tighter bound on $\|\mathbf{x}^{\tilde{k}_0+1}-\mathbf{x}^{\tilde{k}_0}\|$ given in Lemma 10, we have the following corollary, which gives a tighter bound on the gradient tracking error $\|\mathbf{g}^{\tilde{k}_0+1}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0+1}\|$.

Corollary 3. *Under the setting of Lemma 10, we have*

$$\begin{aligned}&\frac{1}{L_1}\|\mathbf{g}^{\tilde{k}_0+1}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0+1}\|\\ \le&\mathbf{J}_{21}^{[3]}\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|+\frac{\mathbf{J}_{22}^{[3]}}{L_1}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\|+\mathbf{J}_{23}^{[3]}\sqrt{n}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|.\end{aligned}$$

Proof. We substitute (44) into (26) and complete the proof. □

Step III: The following corollary bounds $\|\overline{\mathbf{x}}^{k+1}-x^*\|$ based on the locally quadratic convergence of the centralized Newton's method.

Corollary 4. *Under the setting of Lemma 10, we have*

$$\begin{aligned}&\|\overline{\mathbf{x}}^{\tilde{k}_0+1}-x^*\|\\ \le&\mathbf{J}_{33}^{[3]}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|+\mathbf{J}_{31}^{[3]}\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|+\frac{\mathbf{J}_{32}^{[3]}}{L_1}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\|.\end{aligned}$$

Proof. See Supplementary XI. □

Combining Corollaries 2, 3, and 4, we get (22).

To prove (24) from (22), we substitute the parameters satisfying (20) and do algebraic manipulations. Details can be found in the full version [40]. This completes the proof. □

F. Proof of Proposition 6

Proof. According to (17), we have

$$\begin{aligned}&\kappa_F\varrho^k+\vartheta^k\\ =&\frac{\kappa_F}{M_1}\left(\frac{1}{\sqrt{n}}L_2\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\|+\frac{1}{\sqrt{n}}\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F+\mu c_k\right)\\ &+\frac{L_2}{2\mu}\|\overline{\mathbf{x}}^k-x^*\|\\ \le&\frac{\kappa_F}{M_1\sqrt{n}}u_2^k+\frac{\kappa_F L_2}{M_1\sqrt{n}}u_3^k+\frac{1}{40}\sigma^{\frac{m}{2}} \qquad (45)\end{aligned}$$

for all $k\ge 0$, where the inequality holds because $c_k\le\frac{M_1\sigma^{m/2}}{40\mu\kappa_F}$, $\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F\le u_2^k$, and

$$\frac{\kappa_F}{M_1}\frac{1}{\sqrt{n}}L_2\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\|+\frac{L_2}{2\mu}\|\overline{\mathbf{x}}^k-x^*\|\le\frac{\kappa_F L_2}{M_1\sqrt{n}}u_3^k.$$

On the other hand, it is worth noting that (34) holds for any $k\ge 0$. With Theorem 1, we know that (28) holds for any $k\ge 0$. Thus, by substituting $\alpha=1$ into (28), we have

$$\begin{aligned}\|\mathbf{x}^{k+1}-\mathbf{x}^k\|\le&\left(2+\frac{2L_1}{M_1}\right)\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\|\\ &+\frac{2}{M_1}\|\mathbf{g}^k-\mathbf{W}^\infty\mathbf{g}^k\|+\frac{2\sqrt{n}}{M_1}\|\nabla F(\overline{\mathbf{x}}^k)\|\le\frac{4L_1}{M_1}u_3^k,\end{aligned}$$

where the last inequality holds because $\|\nabla F(\overline{\mathbf{x}}^k)\|\le L_1\|\overline{\mathbf{x}}^k-x^*\|$. Define

$$A_k\triangleq u_2^k+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}u_3^k.$$

Then, (45) implies that

$$\kappa_F\varrho^k+\vartheta^k\le\frac{\kappa_F}{M_1\sqrt{n}}A_k+\frac{1}{40}\sigma^{\frac{m}{2}} \qquad (46)$$

for all $k\ge 0$. The motivation behind the definition of the sequence $\{A_k\}_{k\ge0}$ is given as follows. If Proposition 5 holds at time step k, i.e., $u_3^{k+1}\le\sigma^{\frac{m}{2}}u_3^k$, then we have

$$\begin{aligned}A_{k+1}=&u_2^{k+1}+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}u_3^{k+1}\\ \le&u_2^k+\frac{5L_2}{4}\cdot\frac{4L_1}{M_1}u_3^k+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}\sigma^{m/2}u_3^k\\ =&u_2^k+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}u_3^k=A_k.\end{aligned}$$

Here, we use $u_2^{k+1}\le u_2^k+\frac{5L_2}{4}\cdot\frac{4L_1}{M_1}u_3^k$, which is derived from (15) and the condition that $\gamma\le\frac{\delta^2(1-\sigma)}{50}$.

Now, we are ready to prove that (19) and (23) hold for all $k\ge0$ by induction. First, we show that (19) and (23) hold at time step K. Since $F(\overline{\mathbf{x}}^k)-F(x^*)\ge\frac{\mu}{2}\|\overline{\mathbf{x}}^k-\mathbf{x}^*\|^2$, we know that $\mathbf{q}_1^K\ge\frac{1}{2\kappa_F}(\mathbf{q}_3^K)^2$. Thus, we have

$$\begin{aligned}(u_3^K)^2\le&\left(1,\sigma^{-\frac{m}{2}},0.25\sigma^{-\frac{3m}{2}}\right)(\mathbf{q}_3^K)^2\\ \le&2\kappa_F\left(1,\sigma^{-\frac{m}{2}},0.25\sigma^{-\frac{3m}{2}}\right)\mathbf{q}_1^K\\ \le&\frac{100\kappa_F\sigma^{-\frac{5m}{2}}}{(1-\sigma^2)^2}u_1^K,\end{aligned}$$

which implies that

$$\begin{aligned}A_K=&u_2^K+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}u_3^K\\ \le&u_2^K+\frac{5L_1L_2}{M_1(1-\sigma^{m/2})}\cdot\frac{10\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{1-\sigma^2}\sqrt{u_1^K}\\ \le&\tilde{u}_2^0\phi^K+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\left(1-\frac{\mu\alpha}{2M_2}\right)^{\frac{K}{2}}\sqrt{u_1^0}\\ \le&\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)\phi^K.\end{aligned}$$

Here, the inequality holds because $\left(1-\frac{\mu\alpha}{2M_2}\right)^{\frac{1}{2}}\le\phi$.

Further, with (46), we have

$$\kappa_F\varrho^K+\vartheta^K\le\frac{\kappa_F}{M_1\sqrt{n}}A_K+\frac{1}{40}\sigma^{\frac{m}{2}}.$$

Thus, condition (23) holds at time step K if

$$\begin{aligned}\frac{\kappa_F}{M_1}A_K &\le \frac{\kappa_F}{M_1}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)\phi^K\\ &\le \frac{1}{40}\sigma^{\frac{m}{2}},\end{aligned} \tag{47}$$

which is equivalent to

$$\begin{aligned}K\ge&\frac{\log\frac{\sigma^{m/2}}{\frac{40\kappa_F}{M_1}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}}{\log\phi}\\ =&\frac{\frac{m}{2}\log\sigma-\log\frac{40\kappa_F}{M_1\sqrt{n}}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}{\log\phi}.\end{aligned} \tag{48}$$

Besides, based on (41) and the definition of A_k, we have

$$\begin{aligned}&\|H_i^K-\nabla^2F(\overline{\mathbf{x}}^K)\|_F\\ \le&\|\mathbf{H}^K-\mathbf{W}^\infty\mathbf{H}^K\|_F+\frac{L_2}{\sqrt{n}}\|\mathbf{x}^K-\mathbf{W}^\infty\mathbf{x}^K\|_F\\ \le&u_2^K\le A_K\le\frac{M_1}{40}=\frac{\mu}{41},\end{aligned} \tag{49}$$

where we use (47) in the last inequality. Then, (49) implies that

$$M_1I_d=(u-\frac{\mu}{41})I_d\le H_i^K\le(L_1+\frac{\mu}{41})I_d=M_2I_d.$$

Thus, both (19) and (23) hold at time step K.

Second, assume that (19) and (23) hold for $K,\dots,k-1$. To complete the mathematical induction, it remains to show that both (19) and (23) hold at time step k, which can be done with basic algebraic manipulations. Please see the full version [40] for details.

Finally, by substituting $M_1=\frac{40\mu}{41}$ into (48), we get

$$K\ge\frac{\frac{m}{2}\log\sigma-\log\frac{41\kappa_F}{\mu\sqrt{n}}\left(\tilde{u}_2^0+\frac{52L_2\kappa_F\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}{\log\phi}$$

and complete the proof. □

G. Proof of Corollary 1

The result (22) gives an one-step descent that holds for any $m\ge1$, and we can directly replace m with m_k since $m_k\ge m$. To prove Corollary 1, it remains to show that condition (23) still holds for m_k. In other words, if we can prove that

$$\kappa_F\varrho^k+\vartheta^k\le\frac{1}{20}\sigma^{\frac{m_k}{2}}, \tag{50}$$

then we have $u_3^{k+1}\le\sigma^{\frac{m_k}{2}}u_3^k$, which is the desired result. Considering that $m_k\ge m$, which implies smaller consensus errors, we conclude that Propositions 2–4 still hold for any k. Thus, similar to (45) and (46), if we choose $c_k\le\frac{M_1\sigma^{m_k/2}}{40\mu\kappa_F}$, then we have

$$\begin{aligned}\kappa_F\varrho^k+\vartheta^k&\le\frac{\kappa_F}{M_1\sqrt{n}}u_2^k+\frac{\kappa_FL_2}{M_1\sqrt{n}}u_3^k+\frac{1}{40}\sigma^{\frac{m_k}{2}}\\ &\le\frac{\kappa_F}{M_1\sqrt{n}}A_k+\frac{1}{40}\sigma^{\frac{m_k}{2}}.\end{aligned}$$

Using a similar derivation as in (47), condition (50) holds at time step k if

$$\begin{aligned}\frac{\kappa_F}{M_1}A_k&\le\frac{\kappa_F}{M_1}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)\phi^k\\ &\le\frac{1}{40}\sigma^{\frac{m_k}{2}},\end{aligned}$$

which is equivalent to

$$\begin{aligned}k\ge&\frac{\log\frac{\sigma^{m_k/2}}{\frac{40\kappa_F}{M_1}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}}{\log\phi}\\ =&\frac{\frac{m_k}{2}\log\sigma-\log\frac{40\kappa_F}{M_1\sqrt{n}}\left(\tilde{u}_2^0+\frac{50L_1L_2\sqrt{\kappa_F}\sigma^{-\frac{5m}{4}}}{M_1(1-\sigma^{m/2})(1-\sigma^2)}\sqrt{u_1^0}\right)}{\log\phi}\\ =&K+\frac{m_k-m}{2\log\phi}\log\sigma.\end{aligned}$$

Since the above condition is guaranteed by (21), we complete the proof of Corollary 1.

REFERENCES

[1] C. Fang, Z. Yang, and W. U. Bajwa, "Bridge: Byzantine-resilient decentralized gradient descent," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 610–626, 2022.

[2] D. Ciuonzo, S. H. Javadi, A. Mohammadi, and P. S. Rossi, "Bandwidth-constrained decentralized detection of an unknown vector signal via multisensor fusion," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 744–758, 2020.

[3] A. Beznosikov, G. Scutari, A. Rogozin, and A. Gasnikov, "Distributed saddle-point problems under data similarity," in *Advances in Neural Information Processing Systems*, 2021.

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

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

[6] W. Liu, L. Chen, and W. Zhang, "Decentralized federated learning: Balancing communication and computing costs," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 131–143, 2022.

[7] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 482–497, 2014.

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

[9] Q. Ling, W. Shi, G. Wu, and A. Ribeiro, "DLM: Decentralized linearized alternating direction method of multipliers," *IEEE Transactions on Signal Processing*, vol. 63, no. 15, pp. 4051–4064, 2015.

[10] W. Li, Y. Liu, Z. Tian, and Q. Ling, "Communication-censored linearized ADMM for decentralized consensus optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 18–34, 2019.

[11] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, "Exact diffusion for distributed optimization and learning Part I: Algorithm development," *IEEE Transactions on Signal Processing*, vol. 67, no. 3, pp. 708–723, 2018.

[12] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 3, pp. 1245–1260, 2017.

[13] G. Scutari and Y. Sun, "Distributed nonconvex constrained optimization over time-varying digraphs," *Mathematical Programming*, vol. 176, no. 1, pp. 497–544, 2019.

[14] R. Xin, U. A. Khan, and S. Kar, "Fast decentralized nonconvex finite-sum optimization with recursive variance reduction," *SIAM Journal on Optimization*, vol. 32, no. 1, pp. 1–28, 2022.

[15] R. Xin and U. A. Khan, "Distributed heavy-ball: A generalization and acceleration of first-order methods with gradient tracking," *IEEE Transactions on Automatic Control*, vol. 65, no. 6, pp. 2627–2633, 2019.

[16] S. A. Alghunaim, E. Ryu, K. Yuan, and A. H. Sayed, "Decentralized proximal gradient algorithms with linear convergence rates," *IEEE Transactions on Automatic Control*, vol. 66, no. 6, pp. 2787–2794, 2020.

[17] J. Xu, Y. Tian, Y. Sun, and G. Scutari, "Distributed algorithms for composite optimization: Unified framework and convergence analysis," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3555–3570, 2021.

[18] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for smooth and strongly convex distributed optimization in networks," in *International Conference on Machine Learning*, 2017.

[19] H. Li, C. Fang, W. Yin, and Z. Lin, "A sharp convergence rate analysis for distributed accelerated gradient methods," *arXiv preprint arXiv:1810.01053*, 2018.

[20] H. Ye, L. Luo, Z. Zhou, and T. Zhang, "Multi-consensus decentralized accelerated gradient descent," *arXiv preprint arXiv:2005.00797*, 2020.

[21] F. Mansoori and E. Wei, "Superlinearly convergent asynchronous distributed network Newton method," in *IEEE Conference on Decision and Control*, 2017.

[22] A. Mokhtari, Q. Ling, and A. Ribeiro, "Network Newton distributed optimization methods," *IEEE Transactions on Signal Processing*, vol. 65, no. 1, pp. 146–161, 2016.

[23] R. Tutunov, H. Bou-Ammar, and A. Jadbabaie, "Distributed Newton method for large-scale consensus optimization," *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 3983–3994, 2019.

[24] J. Zhang, K. You, and T. Başar, "Distributed adaptive Newton methods with global superlinear convergence," *Automatica*, vol. 138, p. 110156, 2022.

[25] A. Daneshmand, G. Scutari, P. Dvurechensky, and A. Gasnikov, "Newton method over networks is fast up to the statistical precision," in *International Conference on Machine Learning*, 2021, pp. 2398–2409.

[26] E. Berglund, S. Magnússon, and M. Johansson, "Distributed Newton method over graphs: Can sharing of second-order information eliminate the condition number dependence?" *IEEE Signal Processing Letters*, vol. 28, pp. 1180–1184, 2021.

[27] F. Zanella, D. Varagnolo, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton-Raphson consensus for distributed convex optimization," in *IEEE Conference on Decision and Control and European Control Conference*, 2011.

[28] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "DQM: Decentralized quadratically approximated alternating direction method of multipliers," *IEEE Transactions on Signal Processing*, vol. 64, no. 19, pp. 5158–5173, 2016.

[29] M. Eisen, A. Mokhtari, and A. Ribeiro, "A primal-dual quasi-Newton method for exact consensus optimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 23, pp. 5983–5997, 2019.

[30] Y. Sun, G. Scutari, and A. Daneshmand, "Distributed optimization based on gradient tracking revisited: Enhancing convergence rate via surrogation," *SIAM Journal on Optimization*, vol. 32, no. 2, pp. 354–385, 2022.

[31] J. Zhang, Q. Ling, and A. M.-C. So, "A Newton tracking algorithm with exact linear convergence for decentralized consensus optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 7, pp. 346 – 358, 2021.

[32] H. Wei, Z. Qu, X. Wu, H. Wang, and J. Lu, "Decentralized approximate Newton methods for convex optimization on networked systems," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 3, pp. 1489–1500, 2021.

[33] J. Zhang, H. Liu, A. M.-C. So, and Q. Ling, "Variance-reduced stochastic quasi-Newton methods for decentralized learning—Part I: General framework," *arXiv preprint arXiv:2201.07699*, 2022.

[34] ——, "Variance-reduced stochastic quasi-Newton methods for decentralized learning—Part II: Damped limited-memory DFP and BFGS methods," *arXiv preprint arXiv:2201.07733*, 2022.

[35] A. Mokhtari, W. Shi, Q. Ling, and A. Ribeiro, "A decentralized second-order method with exact linear convergence rate for consensus optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 507–522, 2016.

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

[37] M. Safaryan, R. Islamov, X. Qian, and P. Richtárik, "FedNL: Making Newton-type methods applicable to federated learning," *arXiv preprint arXiv:2106.02969*, 2021.

[38] R. Islamov, X. Qian, and P. Richtárik, "Distributed second-order methods with fast rates and compressed communication," in *International Conference on Machine Learning*, 2021, pp. 4617–4628.

[39] X. Liu, Y. Li, R. Wang, J. Tang, and M. Yan, "Linear convergent decentralized optimization with compression," *arXiv preprint arXiv:2007.00232*, 2020.

[40] H. Liu, J. Zhang, A. M.-C. So, and Q. Ling, "A communication-efficient decentralized Newton's method with provably faster convergence," *arXiv preprint arXiv:2210.00184*, 2022.

[41] P. Richtárik, I. Sokolov, and I. Fatkhullin, "Ef21: A new, simpler, theoretically better, and practically faster error feedback," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[42] X. Qian, R. Islamov, M. Safaryan, and P. Richtárik, "Basis matters: Better communication-efficient second order methods for federated learning," *arXiv preprint arXiv:2111.01847*, 2021.

[43] Y.-H. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM Journal on optimization*, vol. 10, no. 1, pp. 177–182, 1999.

[44] Y. Liao, Z. Li, K. Huang, and S. Pu, "A compressed gradient tracking method for decentralized optimization with linear convergence," *IEEE Transactions on Automatic Control*, 2022.

[45] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[46] M.-C. Yue, Z. Zhou, and A. M.-C. So, "On the quadratic convergence of the cubic regularization method under a local error bound condition," *SIAM Journal on Optimization*, vol. 29, no. 1, pp. 904–932, 2019.

[47] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2003, vol. 87.

SUPPLEMENTARY MATERIAL FOR

A Communication-Efficient Decentralized Newton's Method with Provably Faster Convergence

I. PROOF OF LEMMA 1

Proof. First, we show that multi-step consensus with m inner loops improves the convergence rate of the consensus error from σ to σ^m. To do this, we compute

$$\mathbf{W}^m = (\mathbf{W} - \mathbf{W}^\infty + \mathbf{W}^\infty)^m = (\mathbf{W} - \mathbf{W}^\infty)^m + \mathbf{W}^\infty,$$

where we use $(\mathbf{W}^\infty)^2 = \mathbf{W}^\infty$ and $\mathbf{W}\mathbf{W}^\infty = \mathbf{W}^\infty\mathbf{W} = \mathbf{W}^\infty$. Thus, we have

$$\|\mathbf{W}^m - \mathbf{W}^\infty\| = \|\mathbf{W} - \mathbf{W}^\infty\|^m \leq \sigma^m.$$

Then, $\mathbf{x}^{k+1} = \mathbf{W}^m(\mathbf{x}^k - \alpha\mathbf{d}^k)$ implies that

$$\begin{aligned}\mathbf{x}^{k+1} - \mathbf{W}^\infty\mathbf{x}^{k+1} &= (\mathbf{W}^m - \mathbf{W}^\infty)(\mathbf{x}^k - \alpha\mathbf{d}^k) \\ &= (\mathbf{W}^m - \mathbf{W}^\infty)\big(\mathbf{x}^k - \alpha\mathbf{d}^k - \mathbf{W}^\infty(\mathbf{x}^k - \alpha\mathbf{d}^k)\big).\end{aligned}$$

Taking the norm $\|\cdot\|$ on both sides of the above equality and using the triangle inequality, we get (25).

Second, according to $\mathbf{g}^{k+1} = \mathbf{W}^m(\mathbf{g}^k + \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k))$, we have

$$\begin{aligned}&\mathbf{g}^{k+1} - \mathbf{W}^\infty\mathbf{g}^{k+1} \\ =&(\mathbf{W}^m - \mathbf{W}^\infty)\big(\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k + \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k)\big).\end{aligned}$$

Taking the norm $\|\cdot\|$ on both sides of the above inequality and using the triangle inequality and Assumption 2, we get (26) and complete the proof. □

II. PROOF OF LEMMA 2

Proof. According to (5), we have $\overline{\mathbf{g}}^k = \overline{\nabla f}(\mathbf{x}^k)$. Under Assumption 2, we have

$$\|\overline{\mathbf{g}}^k - \nabla F(\overline{\mathbf{x}}^k)\| \leq \frac{L_1}{\sqrt{n}}\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\|, \tag{51}$$

which implies that

$$\begin{aligned}\|\mathbf{g}^k\| \leq& \|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\| + \sqrt{n}\|\overline{\mathbf{g}}^k\| \\ \leq& \|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\| + \sqrt{n}\|\overline{\mathbf{g}}^k - \nabla F(\overline{\mathbf{x}}^k)\| + \sqrt{n}\|\nabla F(\overline{\mathbf{x}}^k)\| \\ \leq& \|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\| + L_1\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\| + \sqrt{n}\|\nabla F(\overline{\mathbf{x}}^k)\|.\end{aligned} \tag{52}$$

This inequality gives (27).

If (9) holds for a certain k_0, according to the fact that $(\mathrm{diag}\{H_i^{k+1}\} + MI_{nd})\mathbf{d}^{k+1} = \mathbf{g}^{k+1} + \mathbf{r}^{k+1}$, we have

$$\|\mathbf{d}^{k_0} - \mathbf{W}^\infty\mathbf{d}^{k_0}\| \leq \|\mathbf{d}^{k_0}\| \leq \frac{\|\mathbf{g}^{k_0} + \mathbf{r}^{k_0}\|}{M_1} \leq \frac{2\|\mathbf{g}^{k_0}\|}{M_1}, \tag{53}$$

where we use $c_k \leq 1$ in the last inequality and the first inequality is given for later use (see (55)). According to $\mathbf{x}^{k+1} = \mathbf{W}^m(\mathbf{x}^k - \alpha\mathbf{d}^k)$, we have

$$\begin{aligned}\|\mathbf{x}^{k_0+1} - \mathbf{x}^{k_0}\| \leq& \|(\mathbf{W}^m - I_{nd})(\mathbf{x}^{k_0} - \mathbf{W}^\infty\mathbf{x}^{k_0})\| + \alpha\|\mathbf{d}^{k_0}\| \\ \leq& 2\|\mathbf{x}^{k_0} - \mathbf{W}^\infty\mathbf{x}^{k_0}\| + \frac{2\alpha}{M_1}\|\mathbf{g}^{k_0}\|,\end{aligned} \tag{54}$$

where we substitute (53) in the last inequality. By substituting (52) into (54), we get (28) and complete the proof. □

III. PROOF OF LEMMA 3

Proof. With (25), we have

$$\begin{aligned}&\|\mathbf{x}^{k_0+1} - \mathbf{W}^\infty\mathbf{x}^{k_0+1}\|^2 \\ \leq& \sigma^{2m}\big((1+\eta_1)\|\mathbf{x}^{k_0} - \mathbf{W}^\infty\mathbf{x}^{k_0}\|^2 \\ &+ \left(1 + \frac{1}{\eta_1}\right)\alpha^2\|\mathbf{d}^{k_0} - \mathbf{W}^\infty\mathbf{d}^{k_0}\|^2\big) \\ \leq& \frac{\sigma^{2m-2}(1+\sigma^2)}{2}\|\mathbf{x}^{k_0} - \mathbf{W}^\infty\mathbf{x}^{k_0}\|^2 + \frac{8\sigma^{2m}\alpha^2}{(1-\sigma^2)M_1^2}\|\mathbf{g}^{k_0}\|^2,\end{aligned} \tag{55}$$

where the first inequality holds for any $\eta_1 > 0$ due to Young's inequality and the second inequality holds by setting $\eta_1 = \frac{1-\sigma^2}{2\sigma^2}$ and substituting into (53).

Further, according to (27), we have

$$\begin{aligned}&\|\mathbf{g}^k\|^2 \\ \leq& 4\|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\|^2 + 2L_1^2\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\|^2 + 4n\|\nabla F(\overline{\mathbf{x}}^k)\|^2 \\ \leq& 4\|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\|^2 + 2L_1^2\|\mathbf{x}^k - \mathbf{W}^\infty\mathbf{x}^k\|^2 \\ &+ 8L_1 n\big(F(\overline{\mathbf{x}}^k) - F(x^*)\big)\end{aligned} \tag{56}$$

for all $k \geq 0$, where the last inequality holds since

$$\|\nabla F(\overline{\mathbf{x}}^k)\|^2 \leq 2L_1\left(F(\overline{\mathbf{x}}^k) - F(x^*)\right), \tag{57}$$

whose proof can be found in [47, Theorem 2.1.5]. By substituting (56) into (55), we get

$$\begin{aligned}&\|\mathbf{x}^{k_0+1} - \mathbf{W}^\infty\mathbf{x}^{k_0+1}\|^2 \\ \leq& \sigma^{2m-2}\left(\frac{1+\sigma^2}{2} + \frac{16\sigma^2\alpha^2L_1^2}{(1-\sigma^2)M_1^2}\right)\|\mathbf{x}^{k_0} - \mathbf{W}^\infty\mathbf{x}^{k_0}\|^2 \\ &+ \frac{32\sigma^{2m}\alpha^2L_1^2}{(1-\sigma^2)M_1^2}\cdot\frac{1}{L_1^2}\|\mathbf{g}^{k_0} - \mathbf{W}^\infty\mathbf{g}^{k_0}\|^2 \\ &+ \frac{64\sigma^{2m}\alpha^2L_1^2}{(1-\sigma^2)M_1^2}\cdot\frac{n}{L_1}\left(F(\overline{\mathbf{x}}^{k_0}) - F(x^*)\right).\end{aligned} \tag{58}$$

To get (29), the remaining is to substitute $\alpha \leq \frac{M_1^2(1-\sigma^2)^3}{100L_1M_2\sigma^{m-1}}$ into (58) and do algebraic manipulations. Details can be found in the full version [40]. This completes the proof. □

IV. PROOF OF LEMMA 4

Proof. With (26), we have

$$\begin{aligned}&\|\mathbf{g}^{k+1} - \mathbf{W}^\infty\mathbf{g}^{k+1}\|^2 \\ \leq& \sigma^{2m}\left((1+\eta_2)\|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\|^2 + \left(1 + \frac{1}{\eta_2}\right)L_1^2\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2\right) \\ \leq& \frac{\sigma^{2m-2}(1+\sigma^2)}{2}\|\mathbf{g}^k - \mathbf{W}^\infty\mathbf{g}^k\|^2 + \frac{2\sigma^{2m}L_1^2}{1-\sigma^2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2\end{aligned} \tag{59}$$

for all $k \geq 0$, where the first inequality holds for any $\eta_2 > 0$ due to Young's inequality and the second inequality holds by setting $\eta_2 = \frac{1-\sigma^2}{2\sigma^2}$.

Further, according to (28), we have

$$\begin{aligned}
&\|\mathbf{x}^{k_0+1}-\mathbf{x}^{k_0}\|^2\\
\leq&2\left(2+\frac{2\alpha L_1}{M_1}\right)^2\|\mathbf{x}^{k_0}-\mathbf{W}^{\infty}\mathbf{x}^{k_0}\|^2\\
&+\frac{16\alpha^2}{M_1^2}\|\mathbf{g}^{k_0}-\mathbf{W}^{\infty}\mathbf{g}^{k_0}\|^2+\frac{16\alpha^2 n}{M_1^2}\|\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\\
\leq&2\left(2+\frac{2\alpha L_1}{M_1}\right)^2\|\mathbf{x}^{k_0}-\mathbf{W}^{\infty}\mathbf{x}^{k_0}\|^2\\
&+\frac{16\alpha^2}{M_1^2}\|\mathbf{g}^{k_0}-\mathbf{W}^{\infty}\mathbf{g}^{k_0}\|^2+\frac{32\alpha^2 L_1}{M_1^2}n(F(\overline{\mathbf{x}}^{k_0})-F(x^*)),
\end{aligned}\tag{60}$$

where we substitute (57) in the last inequality. By substituting (60) into (59), we have

$$\begin{aligned}
&\frac{1}{L_1^2}\|\mathbf{g}^{k_0+1}-\mathbf{W}^{\infty}\mathbf{g}^{k_0+1}\|^2\\
\leq&\sigma^{2m-2}\left(\frac{1+\sigma^2}{2}+\frac{32\alpha^2\sigma^2L_1^2}{(1-\sigma^2)M_1^2}\right)\cdot\frac{1}{L_1^2}\|\mathbf{g}^{k_0}-\mathbf{W}^{\infty}\mathbf{g}^{k_0}\|^2\\
&+\frac{2\sigma^{2m}}{1-\sigma^2}\cdot 2\left(2+\frac{2\alpha L_1}{M_1}\right)^2\|\mathbf{x}^{k_0}-\mathbf{W}^{\infty}\mathbf{x}^{k_0}\|^2\\
&+\frac{2\sigma^{2m}}{1-\sigma^2}\cdot\frac{32\alpha^2L_1^2}{M_1^2}\cdot\frac{n}{L_1}\left(F(\overline{\mathbf{x}}^{k_0})-F(x^*)\right).
\end{aligned}\tag{61}$$

To get (30), the remaining is to substitute $\alpha\leq\frac{M_1^2(1-\sigma^2)^3}{100L_1M_2\sigma^{m-1}}$ into (61) and do algebraic manipulations. Detail can be found in the full version [40]. This completes the proof. □

V. Proof of Lemma 5

Proof. Let us denote $B_i^k=(H_i^k+MI_d)^{-1}$ and $\overline{\mathbf{B}}^k=\frac{1}{n}\sum_{i=1}^n B_i^k$. Since $\overline{\mathbf{x}}^{k+1}=\overline{\mathbf{x}}^k-\alpha\overline{\mathbf{d}}^k$, under Assumption 2 and (9), we have

$$\begin{aligned}
&F(\overline{\mathbf{x}}^{k_0+1})\\
\leq&F(\overline{\mathbf{x}}^{k_0})-\alpha\left\langle\nabla F(\overline{\mathbf{x}}^{k_0}),\overline{\mathbf{d}}^{k_0}\right\rangle+\frac{L_1\alpha^2}{2}\|\overline{\mathbf{d}}^{k_0}\|^2\\
\leq&F(\overline{\mathbf{x}}^{k_0})-\alpha\left\langle\nabla F(\overline{\mathbf{x}}^{k_0}),\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\right\rangle\\
&-\alpha\left\langle\nabla F(\overline{\mathbf{x}}^{k_0}),\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\right\rangle\\
&+L_1\alpha^2\left(\|\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2+\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\right)\\
\leq&F(\overline{\mathbf{x}}^{k_0})-\frac{\alpha}{M_2}\|\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\\
&+\alpha\left(\frac{1}{4M_2}\|\nabla F(\overline{\mathbf{x}}^{k_0})\|^2+M_2\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\right)\\
&+L_1\alpha^2\left(\frac{1}{M_1^2}\|\nabla F(\overline{\mathbf{x}}^{k_0})\|^2+\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\right)\\
=&F(\overline{\mathbf{x}}^{k_0})-\left(\frac{3\alpha}{4M_2}-\frac{L_1\alpha^2}{M_1^2}\right)\|\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\\
&+(M_2\alpha+L_1\alpha^2)\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2,
\end{aligned}$$

where we use $\frac{1}{M_2}\leq\|\overline{\mathbf{B}}^{k_0}\|\leq\frac{1}{M_1}$ in the last inequality. Next, we bound $\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2$. With (7), we have

$$\begin{aligned}
\overline{\mathbf{d}}^k=&\frac{1}{n}\sum_{i=1}^n B_i^k(g_i^k+r_i^k)\\
=&\frac{1}{n}\sum_{i=1}^n B_i^k(g_i^k-\overline{\mathbf{g}}^k)+\overline{\mathbf{B}}^k\overline{\mathbf{g}}^k+\frac{1}{n}\sum_{i=1}^n B_i^k r_i^k\\
=&\frac{1}{n}\sum_{i=1}^n(B_i^k-\frac{1}{2M_1}I_d)(g_i^k-\overline{\mathbf{g}}^k)+\overline{\mathbf{B}}^k\overline{\mathbf{g}}^k+\frac{1}{n}\sum_{i=1}^n B_i^k r_i^k
\end{aligned}$$

for all k, which implies that

$$\begin{aligned}
&\|\overline{\mathbf{d}}^{k_0}-\overline{\mathbf{B}}^{k_0}\nabla F(\overline{\mathbf{x}}^{k_0})\|^2\\
=&\Big\|\frac{1}{n}\sum_{i=1}^n(B_i^{k_0}-\frac{1}{2M_1}I_d)(g_i^{k_0}-\overline{\mathbf{g}}^{k_0})\\
&+\frac{1}{n}\sum_{i=1}^n B_i^{k_0}r_i^{k_0}+\overline{\mathbf{B}}^{k_0}(\overline{\mathbf{g}}^{k_0}-\nabla F(\overline{\mathbf{x}}^{k_0}))\Big\|^2\\
\leq&4\Big\|\frac{1}{n}\sum_{i=1}^n(B_i^{k_0}-\frac{1}{2M_1}I_d)(g_i^{k_0}-\overline{\mathbf{g}}^{k_0})\Big\|^2\\
&+4\|\overline{\mathbf{B}}^{k_0}(\overline{\mathbf{g}}^{k_0}-\nabla F(\overline{\mathbf{x}}^{k_0}))\|^2+2\Big\|\frac{1}{n}\sum_{i=1}^n B_i^{k_0}r_i^{k_0}\Big\|^2\\
\leq&\frac{4}{n}\sum_{i=1}^n\left\|\left(B_i^{k_0}-\frac{1}{2M_1}I_d\right)(g_i^{k_0}-\overline{\mathbf{g}}^{k_0})\right\|^2\\
&+\frac{4}{M_1^2}\|\overline{\mathbf{g}}^{k_0}-\nabla F(\overline{\mathbf{x}}^{k_0})\|^2+\frac{2}{nM_1^2}\sum_{i=1}^n\|r_i^{k_0}\|^2\\
\leq&\frac{1}{nM_1^2}(\|\mathbf{g}^{k_0}-\mathbf{W}^{\infty}\mathbf{g}^{k_0}\|^2+4L_1^2\|\mathbf{x}^{k_0}-\mathbf{W}^{\infty}\mathbf{x}_{k_0}\|^2+2c_{k_0}^2\|\mathbf{g}^{k_0}\|^2),
\end{aligned}\tag{63}$$

where we use $\|\overline{\mathbf{B}}^{k_0}\|\leq\frac{1}{M_1}$, $\|B_i^{k_0}-\frac{1}{2M_1}I_d\|\leq\frac{1}{2M_1}$, and (51). By substituting (63) into (62), we have

$$\begin{aligned}
&F(\overline{\mathbf{x}}^{k_0+1})-F(x^*)\\
\leq&\left(1-2\mu\left(\frac{3\alpha}{4M_2}-\frac{L_1\alpha^2}{M_1^2}\right)\right)\left(F(\overline{\mathbf{x}}^{k_0})-F(x^*)\right)\\
&+\frac{M_2\alpha+L_1\alpha^2}{nM_1^2}\left(\|\mathbf{g}^{k_0}-\mathbf{W}^{\infty}\mathbf{g}^{k_0}\|^2\right.\\
&\left.+4L_1^2\|\mathbf{x}^{k_0}-\mathbf{W}^{\infty}\mathbf{x}_{k_0}\|^2+2c_{k_0}^2\|\mathbf{g}^{k_0}\|^2\right),
\end{aligned}\tag{64}$$

where we use the fact that $\|\nabla F(\overline{\mathbf{x}}^k)\|^2\geq 2\mu(F(\overline{\mathbf{x}}^k)-F(x^*))$ under Assumption 3. Further, according to (27), we have

$$\begin{aligned}
&\|\mathbf{g}^k\|^2\\
\leq&(\|\mathbf{g}^k-\mathbf{W}^{\infty}\mathbf{g}^k\|+L_1\|\mathbf{x}^k-\mathbf{W}^{\infty}\mathbf{x}^k\|+\sqrt{n}\|\nabla F(\overline{\mathbf{x}}^k)\|)^2\\
\leq&3\|\mathbf{g}^k-\mathbf{W}^{\infty}\mathbf{g}^k\|^2+3L_1^2\|\mathbf{x}^k-\mathbf{W}^{\infty}\mathbf{x}^k\|^2+3n\|\nabla F(\overline{\mathbf{x}}^k)\|^2\\
\leq&3\|\mathbf{g}^k-\mathbf{W}^{\infty}\mathbf{g}^k\|^2+3L_1^2\|\mathbf{x}^k-\mathbf{W}^{\infty}\mathbf{x}^k\|^2\\
&+6nL_1(F(\overline{\mathbf{x}}^k)-F(x^*))
\end{aligned}\tag{65}$$

for all $k \geq 0$. Substituting (65) into (64), we have

$$\begin{aligned}
&\frac{n}{L_1}\left(F(\overline{\mathbf{x}}^{k_0+1}) - F(x^*)\right)\\
\leq&\left(1-2\mu\left(\frac{3\alpha}{4M_2}-\frac{L_1\alpha^2}{M_1^2}\right)+\frac{12c_{k_0}^2 L_1(M_2\alpha+L_1\alpha^2)}{M_1^2}\right)\\
&\cdot\frac{n}{L_1}(F(\overline{\mathbf{x}}^{k_0})-F(x^*)) \qquad (66)\\
&+\frac{L_1(M_2\alpha+L_1\alpha^2)}{M_1^2}(1+6c_{k_0}^2)\cdot\frac{1}{L_1^2}\|\mathbf{g}^{k_0}-\mathbf{W}^\infty\mathbf{g}^{k_0}\|^2\\
&+\frac{(M_2\alpha+L_1\alpha^2)L_1}{M_1^2}(4+6c_{k_0}^2)\|\mathbf{x}^{k_0}-\mathbf{W}^\infty\mathbf{x}_{k_0}\|^2.
\end{aligned}$$

With $\alpha \leq \min\left\{\frac{M_1^2(1-\sigma^2)^3}{100L_1M_2\sigma^{m-1}}, \frac{M_1^2}{200L_1M_2}\right\}$, we have $M_2\alpha + L_1\alpha^2 \leq 1.01M_2\alpha$ and $c_k \leq \frac{M_1}{4M_2\sqrt{2\kappa_F}}$. To get (31), the remaining is to substitute these inequalities into (66) and do algebraic manipulations. Please refer to full version [40] for details. This completes the proof.

□

VI. Proof of Lemma 6

Proof. According to Assumption 4, we have

$$\begin{aligned}
\|\mathbf{E}^{k+1}\|_F &\leq (1-\delta)\|\mathbf{E}^k+\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F\\
&\leq (1-\delta)\|\mathbf{E}^k\|_F+(1-\delta)\|\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F.
\end{aligned} \qquad (67)$$

This completes the proof. □

VII. Proof of Lemma 7

Proof. According to Algorithm 1, we have

$$\begin{aligned}
&\mathbf{H}^{k+1}-\tilde{\mathbf{H}}^{k+1}\\
=&\mathbf{H}^k-\tilde{\mathbf{H}}^k-\mathcal{Q}(\mathbf{H}^k-\tilde{\mathbf{H}}^k) \qquad (68)\\
&-\gamma(I_{nd}-\mathbf{W})\hat{\mathbf{H}}^k+\nabla^2 f(\mathbf{x}^{k+1})-\nabla^2 f(\mathbf{x}^k).
\end{aligned}$$

Next, we bound the right-hand side of (68). First, according to Assumption 4, we have

$$\left\|\mathbf{H}^k-\tilde{\mathbf{H}}^k-\mathcal{Q}(\mathbf{H}^k-\tilde{\mathbf{H}}^k)\right\|_F \leq (1-\delta)\|\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F.$$

Second, according to Algorithm 1, we have

$$\hat{\mathbf{H}}^k = \mathbf{H}^k+\mathbf{E}^k-\mathbf{E}^{k+1},$$

which implies that

$$\begin{aligned}
&\|(I_{nd}-\mathbf{W})\hat{\mathbf{H}}^k\|_F\\
\leq&\|(I_{nd}-\mathbf{W})\mathbf{H}^k\|_F+\|(I_{nd}-\mathbf{W})(\mathbf{E}^k-\mathbf{E}^{k+1})\|_F \qquad (69)\\
\leq&2\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F+2\|\mathbf{E}^k\|_F+2\|\mathbf{E}^{k+1}\|_F.
\end{aligned}$$

Finally, combining inequalities (68)–(69) and using Assumption 2, we have

$$\begin{aligned}
&\|\mathbf{H}^{k+1}-\tilde{\mathbf{H}}^{k+1}\|_F\\
\leq&(1-\delta)\|\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F+2\gamma\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F\\
&+2\gamma\|\mathbf{E}^k\|_F+2\gamma\|\mathbf{E}^{k+1}\|_F+L_2\|\mathbf{x}^{k+1}-\mathbf{x}^k\|\\
\leq&\left(1-\delta+2\gamma(1-\delta)\right)\|\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F+4\gamma\|\mathbf{E}^k\|_F\\
&+2\gamma\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F+L_2\|\mathbf{x}^{k+1}-\mathbf{x}^k\|,
\end{aligned}$$

where we use (67) in the last inequality. This gives (32) and completes the proof.

□

VIII. Proof of Lemma 8

Proof. According to (6), we have

$$\begin{aligned}
&(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^{k+1}\\
=&(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^k-\gamma(I_{nd}-\mathbf{W})\hat{\mathbf{H}}^k\\
&+(I_{nd}-\mathbf{W}^\infty)(\nabla^2 f(\mathbf{x}^{k+1})-\nabla^2 f(\mathbf{x}^k)) \qquad (70)\\
=&(I_{nd}-\mathbf{W}^\infty-\gamma(I_{nd}-\mathbf{W}))\,\mathbf{H}^k-\gamma(I_{nd}-\mathbf{W})(\mathbf{E}^k-\mathbf{E}^{k+1})\\
&+(I_{nd}-\mathbf{W}^\infty)(\nabla^2 f(\mathbf{x}^{k+1})-\nabla^2 f(\mathbf{x}^k)),
\end{aligned}$$

where the first equality holds because $(I_{nd}-\mathbf{W}^\infty)(I_{nd}-\mathbf{W}) = I_{nd}-\mathbf{W}-\mathbf{W}^\infty+\mathbf{W}^\infty = I_{nd}-\mathbf{W}$ and the second equality holds because $\hat{\mathbf{H}}^k = \mathbf{H}^k+\mathbf{E}^k-\mathbf{E}^{k+1}$. For the first term on the right-hand side of (70), we have

$$\begin{aligned}
&\|\left(I_{nd}-\mathbf{W}^\infty-\gamma(I_{nd}-\mathbf{W})\right)\mathbf{H}^k\|_F\\
=&\|(1-\gamma)(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^k+\gamma(\mathbf{W}-\mathbf{W}^\infty)\mathbf{H}^k\|_F \qquad (71)\\
=&\|(1-\gamma)(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^k+\gamma(\mathbf{W}-\mathbf{W}^\infty)(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^k\|_F\\
\leq&(1-\gamma+\gamma\sigma)\|(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^k\|_F.
\end{aligned}$$

By taking the Frobenius norm $\|\cdot\|_F$ on both sides of (70), we have

$$\begin{aligned}
&\|(I_{nd}-\mathbf{W}^\infty)\mathbf{H}^{k+1}\|_F\\
\leq&(1-\gamma(1-\sigma))\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F+2\gamma\|\mathbf{E}^k\|_F\\
&+2\gamma\|\mathbf{E}^{k+1}\|_F+\|\nabla^2 f(\mathbf{x}^{k+1})-\nabla^2 f(\mathbf{x}^k)\|_F\\
\leq&(1-\gamma(1-\sigma))\|\mathbf{H}^k-\mathbf{W}^\infty\mathbf{H}^k\|_F+4\gamma\|\mathbf{E}^k\|_F\\
&+2\gamma(1-\delta)\|\mathbf{H}^k-\tilde{\mathbf{H}}^k\|_F+L_2\|\mathbf{x}^{k+1}-\mathbf{x}^k\|,
\end{aligned}$$

where we use (71) in the first inequality and (67) and Assumption 2 in the second inequality. This gives (33) and completes the proof.

□

IX. Proof of Lemma 9

Proof. With $\overline{\mathbf{d}}^k = \frac{1}{n}\sum_{i=1}^n B_i^k\left(g_i^k+r_i^k\right)$, we have

$$\begin{aligned}
\overline{\mathbf{d}}^k =&\frac{1}{n}\sum_{i=1}^n B_i^k g_i^k+\frac{1}{n}\sum_{i=1}^n B_i^k r_i^k-\frac{1}{n}\sum_{i=1}^n(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}g_i^k\\
&+(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\overline{\mathbf{g}}^k \qquad (72)
\end{aligned}$$

for all $k \geq 0$. Then, we compute

$$\begin{aligned}
&\left\|\overline{\mathbf{d}}^k-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\right\|\\
\leq&\left\|(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}(\overline{\mathbf{g}}^k-\nabla F(\overline{\mathbf{x}}^k))\right\|+\left\|\overline{\mathbf{d}}^k-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\overline{\mathbf{g}}^k\right\|\\
=&\left\|(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}(\overline{\mathbf{g}}^k-\nabla F(\overline{\mathbf{x}}^k))\right\|\\
&+\left\|\frac{1}{n}\sum_{i=1}^n B_i^k g_i^k+\frac{1}{n}\sum_{i=1}^n B_i^k r_i^k-\frac{1}{n}\sum_{i=1}^n(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}g_i^k\right\|\\
\leq&\frac{L_1}{\mu\sqrt{n}}\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\|+\frac{1}{n}\sum_{i=1}^n\left\|B_i^k-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\right\|\|g_i^k\|\\
&+\frac{1}{n}\sum_{i=1}^n\|B_i^k\|c_k\|g_i^k\| \qquad (73)
\end{aligned}$$

for all $k \geq 0$, where we use (72) in the equality and $\overline{\mathbf{g}}^k = \frac{1}{n}\sum_{i=1}^n f_i(x_i^k)$ in the last inequality. To bound the second

term on the right-hand side of (73), with the fact that $\|A^{-1}-B^{-1}\| \leq \|A^{-1}\|\|B^{-1}\|\|A-B\|$, we have

$$\begin{aligned}&\left\|(H_i^{\tilde{k}_0})^{-1}-(\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0}))^{-1}\right\| \\ \leq&\frac{1}{\mu M_1}\left\|H_i^{\tilde{k}_0}-\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0})\right\| \\ \leq&\frac{1}{\mu M_1}\Big(\left\|H_i^{\tilde{k}_0}-\overline{\mathbf{H}}^{\tilde{k}_0}\right\|+\left\|\overline{\mathbf{H}}^{\tilde{k}_0}-\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0})\right\|\Big),\end{aligned} \tag{74}$$

where we use $B_i^{\tilde{k}_0}=(H_i^{\tilde{k}_0})^{-1}$ and $M_1 I_d \preceq H_i^{\tilde{k}_0} \preceq M_2 I_d$.

Next, we bound $\left\|\overline{\mathbf{H}}^k-\nabla^2 F(\overline{\mathbf{x}}^k)\right\|$ on the right-hand side of (74). According to (6) and the initialization $H_i^0=\nabla^2 f_i(\mathbf{x}^0)$, we know that

$$\overline{\mathbf{H}}^k=\overline{\nabla^2 f}(\mathbf{x}^k) \tag{75}$$

for all $k\geq 0$. Then, according to Assumption 2, (75) implies that

$$\|\overline{\mathbf{H}}^k-\nabla^2 F(\overline{\mathbf{x}}^k)\| \leq \frac{L_2}{\sqrt{n}}\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\| \tag{76}$$

for all $k\geq 0$. Substituting (76) into (74), we have

$$\begin{aligned}&\left\|(H_i^{\tilde{k}_0})^{-1}-(\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0}))^{-1}\right\| \\ \leq&\frac{1}{\mu M_1}\Big(\left\|H_i^{\tilde{k}_0}-\overline{\mathbf{H}}^{\tilde{k}_0}\right\|+\frac{L_2}{\sqrt{n}}\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|\Big).\end{aligned} \tag{77}$$

Substituting (77) into (73), we have

$$\begin{aligned}&\left\|\overline{\mathbf{d}}^{\tilde{k}_0}-(\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0}))^{-1}\nabla F(\overline{\mathbf{x}}^{\tilde{k}_0})\right\| \\ \leq&\frac{L_1}{\mu\sqrt{n}}\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|+\frac{1}{n}\sum_{i=1}^n\|B_i^{\tilde{k}_0}\|c_{\tilde{k}_0}\|g_i^{\tilde{k}_0}\| \\ &+\frac{1}{\mu M_1 n}\sum_{i=1}^n\|H_i^{\tilde{k}_0}-\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0})\|\|g_i^{\tilde{k}_0}\| \\ \leq&\frac{L_1}{\mu\sqrt{n}}\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|+\frac{1}{\mu M_1 n}\Big(L_2\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\| \\ &+\|\mathbf{H}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{H}^{\tilde{k}_0}\|_F+\mu c_{\tilde{k}_0}\sqrt{n}\Big)\|\mathbf{g}^{\tilde{k}_0}\|.\end{aligned} \tag{78}$$

In addition, following similar steps as in the derivation of (78), we have

$$\begin{aligned}&\|\mathbf{d}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{d}^{\tilde{k}_0}\| \\ \leq&\left\|(I_{nd}-\mathbf{W}^\infty)\left(\mathbf{d}^{\tilde{k}_0}-(\mathrm{diag}\{\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0})\})^{-1}\mathbf{g}^{\tilde{k}_0}\right)\right\| \\ &+\left\|(I_{nd}-\mathbf{W}^\infty)(\mathrm{diag}\{\nabla^2 F(\overline{\mathbf{x}}^{\tilde{k}_0})\})^{-1}\mathbf{g}^{\tilde{k}_0}\right\| \\ \leq&\frac{\|\mathbf{g}^{\tilde{k}_0}\|}{\mu M_1\sqrt{n}}\Big(L_2\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\|+\|\mathbf{H}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{H}^{\tilde{k}_0}\|_F \\ &+\mu c_{\tilde{k}_0}\sqrt{n}\Big)+\frac{1}{\mu}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| \\ =&\frac{\varrho^{\tilde{k}_0}}{\mu}\|\mathbf{g}^{\tilde{k}_0}\|+\frac{1}{\mu}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\|.\end{aligned} \tag{79}$$

By substituting (27) into (78) and (79), we get (42) and (43). This completes the proof. □

X. Proof of Lemma 10

Proof. According to $\mathbf{x}^{k+1}=\mathbf{W}^m(\mathbf{x}^k-\alpha\mathbf{d}^k)$, we have

$$\begin{aligned}&\|\mathbf{x}^{k+1}-\mathbf{x}^k\| \\ \leq&\|(\mathbf{W}^m-I_{nd})(\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k)\|+\alpha\|\mathbf{W}^m\mathbf{d}^k\| \\ \leq&2\|\mathbf{x}^k-\mathbf{W}^\infty\mathbf{x}^k\|+\alpha\|\mathbf{W}^m\mathbf{d}^k\|\end{aligned} \tag{80}$$

for all $k\geq 0$. Next, we bound the second term $\|\mathbf{W}^m\mathbf{d}^k\|$ on the right-hand side of (80). According to the triangle inequality, we have

$$\begin{aligned}&\|\mathbf{W}^m\mathbf{d}^k\| \\ \leq&\|\mathbf{W}^m\mathbf{d}^k-\mathbf{W}^\infty\mathbf{d}^k\|+\sqrt{n}\|\overline{\mathbf{d}}^k-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\| \\ &+\sqrt{n}\|(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\|\end{aligned} \tag{81}$$

for all $k\geq 0$. For the third term on the right-hand side of (81), according to the fact that

$$\|\overline{\mathbf{x}}^k-x^*-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\| \leq \frac{L_2}{2\mu}\|\overline{\mathbf{x}}^k-x^*\|^2 \tag{82}$$

holds for all $k\geq 0$, we have

$$\|(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\| \leq \|\overline{\mathbf{x}}^k-x^*\|+\frac{L_2}{2\mu}\|\overline{\mathbf{x}}^k-x^*\|^2$$

for all $k\geq 0$. For the first term on the right-hand side of (81), we have $\|\mathbf{W}^m\mathbf{d}^k-\mathbf{W}^\infty\mathbf{d}^k\| \leq \sigma^m\|\mathbf{d}^k-\mathbf{W}^\infty\mathbf{d}^k\|$ and then use (43) to bound it. For the second term on the right-hand side of (81), we use (42) to bound it. Thus, (81) implies that

$$\begin{aligned}\|\mathbf{W}^m\mathbf{d}^{\tilde{k}_0}\| \leq&\kappa_F(1+2\varrho^{\tilde{k}_0})\|\mathbf{x}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{x}^{\tilde{k}_0}\| \\ &+\kappa_F(\sigma^m+2\varrho^{\tilde{k}_0})\cdot\frac{1}{L_1}\|\mathbf{g}^{\tilde{k}_0}-\mathbf{W}^\infty\mathbf{g}^{\tilde{k}_0}\| \\ &+\left(1+2\kappa_F\varrho^{\tilde{k}_0}+\frac{L_2}{2\mu}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|\right)\sqrt{n}\|\overline{\mathbf{x}}^{\tilde{k}_0}-x^*\|.\end{aligned} \tag{83}$$

Finally, substituting (83) into (80), we get (44) and complete the proof. □

XI. Proof of Corollary 4

Proof. With $\overline{\mathbf{x}}^{k+1}=\overline{\mathbf{x}}^k-\alpha\overline{\mathbf{d}}^k$, we have

$$\begin{aligned}\|\overline{\mathbf{x}}^{k+1}-x^*\| \leq&\|\overline{\mathbf{x}}^k-x^*-\alpha(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\| \\ &+\alpha\|\overline{\mathbf{d}}^k-(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\|\end{aligned} \tag{84}$$

for all $k\geq 0$. With (82), which is given by the centralized Newton's method [45], we get

$$\begin{aligned}&\|\overline{\mathbf{x}}^k-x^*-\alpha(\nabla^2 F(\overline{\mathbf{x}}^k))^{-1}\nabla F(\overline{\mathbf{x}}^k)\| \\ \leq&(1-\alpha)\|\overline{\mathbf{x}}^k-x^*\|+\frac{\alpha L_2}{2\mu}\|\overline{\mathbf{x}}^k-x^*\|^2 \\ =&(1-\alpha+\alpha\vartheta^k)\|\overline{\mathbf{x}}^k-x^*\|\end{aligned} \tag{85}$$

for all $k\geq 0$. By substituting (85) and (42) into (84), we complete the proof. □