A Newton Tracking Algorithm with Exact Linear Convergence for Decentralized Consensus Optimization

Jiaojiao Zhang*, Qing Ling†, and Anthony Man-Cho So*

***Abstract*—This paper considers the problem of decentralized consensus optimization over a network, where each node holds a strongly convex and twice-differentiable local objective function. Our goal is to minimize the sum of the local objective functions and find the exact optimal solution using only local computation and neighboring communication. We propose a novel Newton tracking algorithm, which updates the local variable in each node along a local Newton direction modified with neighboring and historical information. We investigate the connections between the proposed Newton tracking algorithm and several existing methods, including gradient tracking and primal-dual methods. We prove that the proposed algorithm converges to the exact optimal solution at a linear rate. Furthermore, when the iterate is close to the optimal solution, we show that the proposed algorithm requires $O\left(\max\left\{\kappa_f\sqrt{\kappa_g}+\kappa_f^2, \frac{\kappa_g^{3/2}}{\kappa_f}+\kappa_f\sqrt{\kappa_g}\right\}\log\frac{1}{\Delta}\right)$ iterations to find a Δ-optimal solution, where κ_f and κ_g are condition numbers of the objective function and the graph, respectively. Our numerical results demonstrate the efficacy of Newton tracking and validate the theoretical findings.**

I. Introduction

In this paper, we study the problem of decentralized consensus optimization over an undirected and connected network with n nodes, which takes the form

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

Here, $f_i : \mathbb{R}^p \to \mathbb{R}$ is a strongly convex and twice-differentiable function privately owned by node i. Every node aims to obtain an optimal solution x^* to (1) via local computation and communication with its neighbors. Problem (1) arises in various applications, such as resource allocation [1]–[3], smart grid control [4], [5], federated learning [6]–[8], and decentralized machine learning [9]–[12].

Decentralized consensus optimization methods have been extensively studied in the literature. Among the first-order methods, a popular algorithm is decentralized gradient descent (DGD) [13]–[15]. However, DGD has to employ diminishing step sizes to obtain an exact optimal solution. With a fixed step size, DGD converges fast but only to a neighborhood of an optimal solution [15]. There are other first-order methods that use a fixed step size but still converge to an exact optimal solution, including DLM [16], EXTRA [17], exact diffusion [18], NIDS [19], and gradient tracking [20]–[24]. In gradient tracking algorithms, for instance, each node maintains a local estimate of the global gradient descent direction based on neighboring and historical information and uses it to correct the convergence error in DGD. Unification and generalization of several exact decentralized first-order methods are discussed in [25].

Although first-order methods enjoy low per-iteration computational complexity, second-order methods are attractive due to their faster convergence speeds and hence lower communication costs. Some works, such as [26]–[29], consider a penalty function approach, in which the consensus constraint is implicitly enforced by adding a penalty term to the objective function. They then propose second-order methods to tackle the resulting unconstrained formulation. However, these methods can only be proven to converge to a neighborhood of an optimal solution. In essence, there is a tradeoff between convergence speed and solution accuracy in the penalty function approach. To better handle this tradeoff, various second-order methods that operate in the primal-dual domain have been proposed [30]–[32] and can be shown to achieve convergence to the exact optimal solution at a linear rate. The Newton-Raphson consensus method proposed in [33], which operates in the primal domain, also achieves exact linear convergence. However, it requires to exchange both gradient trackers and Hessian trackers in order to estimate the global Newton direction. There are other second-order methods that can achieve superlinear convergence rates, but they typically require much stricter conditions. For instance, the work [34] proposes the distributed averaged quasi-Newton method for a master-slave network, but the initialization is required to be close enough to an optimal solution so as to guarantee local superlinear convergence. The work [35] proposes an algorithm based on Polyak's adaptive Newton method and establishes its global superlinear convergence, but the algorithm needs to run a finite-time set-consensus inner loop in each iteration. Online algorithms with distributed data sources can be found in [36], [37], while this work deals with offline optimization where the local cost, its gradient, and its

Jiaojiao Zhang and Anthony Man-Cho So are with Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong. Qing Ling is with School of Computer Science and Engineering and Guangdong Province Key Laboratory of Computational Science, Sun Yat-Sen University, Guangzhou, Guangdong 510006, China, and also with Pazhou Lab, Guangzhou, Guangdong 510300, China.

A preliminary version of this paper has appeared in the Proceedings of the 59th IEEE Conference on Decision and Control (CDC 2020). Qing Ling is supported in part by NSF China Grant 61973324, Fundamental Research Funds for the Central Universities, and Guangdong Province Key Laboratory of Computational Science under grant 2020B1212060032. Anthony Man-Cho So is supported in part by Hong Kong Research Grants Council (RGC) General Research Fund (GRF) project CUHK 14203218 and CUHK Research Sustainability of Major RGC Funding Schemes project 3133236.

Hessian are known at each node.

In this paper, we propose a novel second-order Newton tracking algorithm, which updates the local variable in each node along a local Newton direction modified with neighboring and historical information. As its name suggests, Newton tracking inherits the idea of gradient tracking, but it can improve the convergence speed of the latter by utilizing second-order information. We investigate the connections between the proposed Newton tracking algorithm and several existing methods, including gradient tracking and primal-dual methods. Under the aforementioned setting of problem (1), we prove that the proposed algorithm converges to the exact optimal solution at a linear rate. Our numerical experiments demonstrate the efficacy of Newton tracking and validate our theoretical findings.

Notation. We use $\mathbf{I} \in \mathbb{R}^{np\times np}$, $I_n \in \mathbb{R}^{n\times n}$, and $I_p \in \mathbb{R}^{p\times p}$ to denote identity matrices of different sizes; $\mathbf{0} \in \mathbb{R}^{np}$ and $0_p \in \mathbb{R}^p$ to denote all-zero vectors of different sizes; $1_n \in \mathbb{R}^n$ to denote the all-one vector; $\lambda_{\max}(\cdot)$, $\lambda_{\min}(\cdot)$, and $\hat{\lambda}_{\min}(\cdot)$ to denote the largest, smallest, and smallest nonzero eigenvalues of a matrix, respectively.

II. Problem Formulation and Algorithm Development

In this section, we rewrite the decentralized consensus optimization problem (1) into an equivalent constrained form and propose the Newton tracking algorithm to solve it.

A. Problem Formulation

Consider a bi-directionally connected network of n nodes, where two nodes are neighbors if they are connected by an edge. Define $\mathcal{N}_i$ as the set that includes the neighbors of node i as well as node i itself. Let $x_i \in \mathbb{R}^p$ be the local copy of the decision variable x that is kept at node i. Since the network is bi-directionally connected, problem (1) is equivalent to

$$\{x_i^*\}_{i=1}^n := \underset{\{x_i\}_{i=1}^n}{\arg\min} \sum_{i=1}^n f_i(x_i), \quad \text{s.t. } x_i = x_j,\ \forall j \in \mathcal{N}_i,\ \forall i. \tag{2}$$

Indeed, the constraint in (2) enforces the consensus condition $x_1 = \cdots = x_n$ for any feasible solution of (2). When the consensus condition is satisfied, the objective functions in (1) and (2) are equivalent, so that the optimal solutions $\{x_i^*\}_{i=1}^n$ of the local problems (2) are all equal to the optimal solution x^* of (1); i.e., $x_1^* = \cdots = x_n^* = x^*$.

B. Algorithm Development

To model the communication process between nodes, we introduce a nonnegative mixing matrix $W \in \mathbb{R}^{n\times n}$, whose (i,j)-th element $w_{ij} \geq 0$ represents the weight assigned to node j by node i. The mixing matrix W is required to satisfy the following assumption, which is standard in the literature.

Assumption 1. *The mixing matrix W is nonnegative, whose (i,j)-th element $w_{ij} \geq 0$ and $w_{ij} = 0$ if and only if $j \notin \mathcal{N}_i$. Further, W is symmetric and doubly stochastic; i.e., $W = W^T$ and $W1_n = 1_n$. The null space of $I_n - W$ is* $\operatorname{span}(1_n)$.

When the underlying network is bi-directionally connected, a mixing matrix W satisfying Assumption 1 can be generated using various techniques, such as those introduced in [38]. According to the Perron-Frobenius theorem, Assumption 1 means that the eigenvalues of W lie in $(-1, 1]$ and W has a single eigenvalue at 1; see, e.g., [39].

At time t of our proposed Newton tracking algorithm, each node i keeps a local copy $x_i^t \in \mathbb{R}^p$ and a vector $u_i^t \in \mathbb{R}^p$ that estimates the negative Newton direction u^t; i.e.,

$$u_i^t \approx u^t \triangleq \left(\sum_{i=1}^n \nabla^2 f_i(\bar{x}^t)\right)^{-1} \left(\sum_{i=1}^n \nabla f_i(\bar{x}^t)\right),$$

where $\bar{x}^t \triangleq \frac{1}{n}\sum_{i=1}^n x_i^t$ is the average of local copies. Each node i updates x_i^{t+1} from x_i^t by moving along the direction $-u_i^t$ with a unit step size. Since it is too expensive to compute the exact Newton direction in a decentralized manner, we propose to estimate the Newton direction by a novel Newton tracking technique. Specifically, our Newton tracking algorithm starts with $x_i^0 = 0_p$ and $u_i^0 = \left(\nabla^2 f_i(0_p) + \epsilon I_p\right)^{-1} \nabla f_i(0_p)$ and then performs the updates

$$x_i^{t+1} = x_i^t - u_i^t, \tag{3}$$

$$u_i^{t+1} = (\nabla^2 f_i(x_i^{t+1}) + \epsilon I_p)^{-1} \Bigg[(\nabla^2 f_i(x_i^t) + \epsilon I_p)u_i^t + \nabla f_i(x_i^{t+1}) - \nabla f_i(x_i^t) + 2\alpha\left(x_i^{t+1} - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^{t+1}\right) - \alpha\left(x_i^t - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^t\right)\Bigg], \tag{4}$$

where $\epsilon > 0$ and $\alpha > 0$ are parameters. Comparing $-u_i^{t+1}$ with the true Newton direction $-u^{t+1}$, we have two observations. First, the exact global Hessian $\sum_{i=1}^n \nabla^2 f_i(\bar{x}^{t+1})$ is replaced by the regularized local Hessian $\nabla^2 f_i(x_i^{t+1}) + \epsilon I_p$. The regularization parameter ϵ is necessary because the local Hessian $\nabla^2 f_i(x_i^{t+1})$ may be unreliable, especially in the beginning stage of the algorithm. Second, the exact gradient $\sum_{i=1}^n \nabla f_i(\bar{x}^t)$ is replaced by three terms that are locally computable. The first term $(\nabla^2 f_i(x_i^t) + \epsilon I_p)u_i^t$ involves the previous local Hessian and the estimated Newton direction. The second term $\nabla f_i(x_i^{t+1}) - \nabla f_i(x_i^t)$ is the difference between the current and previous gradient directions. The third term $2\alpha(x_i^{t+1} - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^{t+1}) - \alpha(x_i^t - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^t)$ extrapolates the current and previous consensus errors. We will give the derivations of (3)-(4) in Section III-B.

Now, let us elaborate on (3)-(4) to better illustrate the idea of Newton tracking. From (4) we have

$$\begin{aligned}&(\nabla^2 f_i(x_i^{t+1}) + \epsilon I_p)u_i^{t+1} \\ =&(\nabla^2 f_i(x_i^t) + \epsilon I_p)u_i^t + \nabla f_i(x_i^{t+1}) - \nabla f_i(x_i^t) \\ &+ 2\alpha\left(x_i^{t+1} - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^{t+1}\right) - \alpha\left(x_i^t - \sum_{j\in\mathcal{N}_i} w_{ij}x_j^t\right).\end{aligned} \tag{5}$$

Summing (5) over $i = 1, \ldots, n$ and invoking the double stochasticity of W, we have

$$\sum_{i=1}^{n} \left(\nabla^2 f_i(x_i^{t+1}) + \epsilon I_p\right) u_i^{t+1} \tag{6}$$
$$= \sum_{i=1}^{n} \left(\nabla^2 f_i(x_i^t) + \epsilon I_p\right) u_i^t + \sum_{i=1}^{n} \left(\nabla f_i(x_i^{t+1}) - \nabla f_i(x_i^t)\right).$$

When the algorithm is initialized such that $\sum_{i=1}^n \nabla f_i(x_i^0) = \sum_{i=1}^n \left(\nabla^2 f_i(x_i^0) + \epsilon I_p\right) u_i^0$, summing (6) from time 0 to time t yields

$$\sum_{i=1}^{n} \left(\nabla^2 f_i(x_i^t) + \epsilon I_p\right) u_i^t = \sum_{i=1}^{n} \nabla f_i(x_i^t).$$

In comparison, the global Newton direction $-u^t$ satisfies

$$\sum_{i=1}^{n} \nabla^2 f_i(\bar{x}^t) u^t = \sum_{i=1}^{n} \nabla f_i(\bar{x}^t).$$

When the local variable pairs $\{(x_i^t, u_i^t)\}_{i=1}^n$ are similar across the nodes, we observe that x_i^t is close to $\bar{x}^t$ and $-u_i^t$ tracks a regularized Newton direction.

For subsequent development, it is desirable to write the updates (3)-(4) in a compact form. Towards that end, define $\mathbf{x} \triangleq [x_1; \ldots; x_n] \in \mathbb{R}^{np}$ and $\mathbf{u} \triangleq [u_1; \ldots; u_n] \in \mathbb{R}^{np}$ as the stacks of local variables. Define the aggregate function $f : \mathbb{R}^{np} \to \mathbb{R}$ as $f(\mathbf{x}) = f(x_1, \ldots, x_n) = \sum_{i=1}^n f_i(x_i)$. The gradient of f at $\mathbf{x}$ is $\nabla f(\mathbf{x}) = [\nabla f_1(x_1); \ldots; \nabla f_n(x_n)] \in \mathbb{R}^{np}$. The Hessian of f at $\mathbf{x}$, denoted by $\nabla^2 f(\mathbf{x}) \in \mathbb{R}^{np \times np}$, is the block diagonal matrix whose i-th diagonal block is $\nabla^2 f_i(x_i)$. Define $\mathbf{H}^t \triangleq \nabla^2 f(\mathbf{x}^t) + \epsilon \mathbf{I} \in \mathbb{R}^{np \times np}$ and $\mathbf{W} \triangleq W \otimes I_p \in \mathbb{R}^{np \times np}$ as the Kronecker product of the weight matrix W and the identity matrix I_p. We can then write (3)-(4) as

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \mathbf{u}^t, \tag{7}$$
$$\mathbf{u}^{t+1} = (\mathbf{H}^{t+1})^{-1}\big[\mathbf{H}^t \mathbf{u}^t + \nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right) + \alpha(\mathbf{I} - \mathbf{W})(2\mathbf{x}^{t+1} - \mathbf{x}^t)\big]. \tag{8}$$

The algorithm is initialized as $\mathbf{x}^0 = \mathbf{0}$ and $\mathbf{u}^0 = (\nabla^2 f(\mathbf{0}) + \epsilon \mathbf{I})^{-1} \nabla f(\mathbf{0})$.

III. Connections with Existing Approaches

This section investigates the connections of the proposed Newton tracking algorithm with several existing approaches, such as gradient tracking and primal-dual methods.

A. Connection with Gradient Tracking

The gradient tracking updates are given by [21]

$$\mathbf{x}^{t+1} = \mathbf{W}\mathbf{x}^t - \alpha \mathbf{y}^t, \tag{9}$$
$$\mathbf{y}^{t+1} = \mathbf{W}\mathbf{y}^t + \nabla f(\mathbf{x}^{t+1}) - \nabla f(\mathbf{x}^t), \tag{10}$$

where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{np}$. To see the connection between gradient tracking and Newton tracking, we first rewrite (9) as $\mathbf{x}^{t+1} = \mathbf{x}^t - [(\mathbf{I} - \mathbf{W})\mathbf{x}^t + \alpha \mathbf{y}^t]$. Then, by defining $\mathbf{r}^t = (\mathbf{I} - \mathbf{W})\mathbf{x}^t + \alpha \mathbf{y}^t \in \mathbb{R}^{np}$, we see that (9)-(10) are equivalent to

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \mathbf{r}^t, \tag{11}$$
$$\mathbf{r}^{t+1} = \mathbf{W}\mathbf{r}^t + \alpha[\nabla f(\mathbf{x}^{t+1}) - \nabla f(\mathbf{x}^t)] + (\mathbf{I} - \mathbf{W})(\mathbf{x}^{t+1} - \mathbf{W}\mathbf{x}^t). \tag{12}$$

Similar to the update of $\mathbf{u}^{t+1}$ in (8), the update of $\mathbf{r}^{t+1}$ in (12) also involves three parts: the previous direction $\mathbf{r}^t$, the difference between current and previous gradient directions $\alpha[\nabla f(\mathbf{x}^{t+1}) - \nabla f(\mathbf{x}^t)]$, and the combination of current and previous consensus errors $(\mathbf{I} - \mathbf{W})(\mathbf{x}^{t+1} - \mathbf{W}\mathbf{x}^t)$. The major difference between $\mathbf{u}^{t+1}$ and $\mathbf{r}^{t+1}$ is that the former utilizes the current and previous Hessians.

B. Connection with Primal-dual Algorithms

The proposed Newton tracking algorithm also has a primal-dual interpretation. Since by assumption the null space of $I_n - W$ is $\operatorname{span}(1_n)$, so is the null space of its square root $(I_n - W)^{\frac{1}{2}}$. Using the relation $(\mathbf{I} - \mathbf{W})^{\frac{1}{2}} = (I_n - W)^{\frac{1}{2}} \otimes I_p$, we see that $(\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{x} = \mathbf{0}$ if and only if $x_1 = \cdots = x_n$. Hence, problem (2) is equivalent to

$$\mathbf{x}^* \triangleq \arg\min_{\mathbf{x}} \ f(\mathbf{x}), \quad \text{s.t. } (\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{x} = \mathbf{0}. \tag{13}$$

The augmented Lagrangian $L(\cdot, \cdot)$ of (13) is given by

$$L(\mathbf{x}, \mathbf{v}) = f(\mathbf{x}) + \langle \mathbf{v}, (\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{x} \rangle + \frac{\alpha}{2} \mathbf{x}^T (\mathbf{I} - \mathbf{W}) \mathbf{x},$$

where $\mathbf{v} \in \mathbb{R}^{np}$ is the dual variable. Therefore, the augmented Lagrangian method for solving (13) can be written as [40]

$$\mathbf{x}^{t+1} = \arg\min_{\mathbf{x}} \ L(\mathbf{x}, \mathbf{v}^k), \tag{14}$$
$$\mathbf{v}^{t+1} = \mathbf{v}^t + \alpha(\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{x}^{t+1}. \tag{15}$$

Despite its simplicity in description, the above method is non-trivial to implement, especially the minimization step in (14). Indeed, given the generality of f, it is unlikely that $\mathbf{x}^{t+1}$ can be given in closed form. Even if f is quadratic and hence $\mathbf{x}^{t+1}$ can be given in closed-form, it cannot be computed in a decentralized manner due to the topology-dependent quadratic term $\frac{\alpha}{2} \mathbf{x}^T (\mathbf{I} - \mathbf{W}) \mathbf{x}$. Motivated by these observations, we propose to replace the functions f and $\mathbf{x} \mapsto \frac{\alpha}{2} \mathbf{x}^T (\mathbf{I} - \mathbf{W}) \mathbf{x}$ in the augmented Lagrangian L by their *quadratic* and *linear* approximations at $\mathbf{x}^t$, respectively, and add the proximal term $\mathbf{x} \mapsto \frac{\epsilon}{2} \|\mathbf{x} - \mathbf{x}^t\|^2$ to the modified augmented Lagrangian. In other words, the update of $\mathbf{x}^{t+1}$ is given by the solution of

$$\min_{\mathbf{x}} \ \left\langle \nabla f(\mathbf{x}^t) + (\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{v}^t + \alpha(\mathbf{I} - \mathbf{W}) \mathbf{x}^t, \mathbf{x} - \mathbf{x}^t \right\rangle + \frac{1}{2} (\mathbf{x} - \mathbf{x}^t)^T \nabla^2 f(\mathbf{x}^t) (\mathbf{x} - \mathbf{x}^t) + \frac{\epsilon}{2} \|\mathbf{x} - \mathbf{x}^t\|^2,$$

which is

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \left(\mathbf{H}^t\right)^{-1} \left[\nabla f\left(\mathbf{x}^t\right) + (\mathbf{I} - \mathbf{W})^{\frac{1}{2}} \mathbf{v}^t + \alpha(\mathbf{I} - \mathbf{W}) \mathbf{x}^t\right]. \tag{16}$$

Now, we show that the updates (16) and (15) initialized by $\mathbf{x}^0 = \mathbf{0}$ and $\mathbf{v}^0 = \mathbf{0}$ are equivalent to the updates (7) and

(8) initialized by $\mathbf{x}^0 = \mathbf{0}$ and $\mathbf{u}^0 = (\nabla^2 f(\mathbf{0}) + \epsilon \mathbf{I})^{-1} \nabla f(\mathbf{0})$. First, observe that both sets of updates give $\mathbf{x}^1 = -(\nabla^2 f(\mathbf{0}) + \epsilon \mathbf{I})^{-1} \nabla f(\mathbf{0})$. Next, using (16), we have

$$\begin{aligned}&\mathbf{H}^t \mathbf{x}^{t+1}\\ &= \mathbf{H}^t \mathbf{x}^t - \left[\nabla f\left(\mathbf{x}^t\right) + (\mathbf{I}-\mathbf{W})^{\frac{1}{2}} \mathbf{v}^t + \alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t\right].\end{aligned}$$

This, together with the dual update (15), implies that

$$\begin{aligned}&\mathbf{H}^{t+1}\mathbf{x}^{t+2} - \mathbf{H}^t \mathbf{x}^{t+1}\\ &= \left[\mathbf{H}^{t+1} - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^{t+1} - \left[\mathbf{H}^t - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^t\\ &\quad - (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}(\mathbf{v}^{t+1}-\mathbf{v}^t) - \left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right]\\ &= \left[\mathbf{H}^{t+1} - 2\alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^{t+1} - \left[\mathbf{H}^t - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^t\\ &\quad - \left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right],\end{aligned}$$

or equivalently,

$$\begin{aligned}&\mathbf{H}^{t+1}\mathbf{x}^{t+2} - \left[\mathbf{H}^{t+1} - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^{t+1}\\ &= \mathbf{H}^t \mathbf{x}^{t+1} - \left[\mathbf{H}^t - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^t - \alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^{t+1}\\ &\quad - \left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right].\end{aligned} \tag{17}$$

Let $\mathbf{s}^t \triangleq \mathbf{H}^t \mathbf{x}^{t+1} - \left[\mathbf{H}^t - \alpha(\mathbf{I}-\mathbf{W})\right]\mathbf{x}^t$. Then, we can rewrite (17) as

$$\mathbf{s}^{t+1} = \mathbf{s}^t - \alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^{t+1} - \left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right]. \tag{18}$$

Moreover, from the definition of $\mathbf{s}^t$, we have

$$\mathbf{x}^{t+1} = \mathbf{x}^t - (\mathbf{H}^t)^{-1}\left[\alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t - \mathbf{s}^t\right]. \tag{19}$$

Upon letting $\mathbf{q}^t \triangleq \alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t - \mathbf{s}^t = \mathbf{H}^t(\mathbf{x}^t - \mathbf{x}^{t+1})$, we can rewrite (19) and (18) as

$$\mathbf{x}^{t+1} = \mathbf{x}^t - (\mathbf{H}^t)^{-1}\mathbf{q}^t, \tag{20}$$

$$\begin{aligned}\mathbf{q}^{t+1} = \mathbf{q}^t &+ \nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\\ &+ \alpha(\mathbf{I}-\mathbf{W})(2\mathbf{x}^{t+1} - \mathbf{x}^t),\end{aligned} \tag{21}$$

respectively. To establish the claimed equivalence, it remains to observe that (20)-(21) corresponds to (7)-(8) with $\mathbf{u}^t = (\mathbf{H}^t)^{-1}\mathbf{q}^t$.

Remark 1. *The exact second-order method (ESOM) introduced in [31] employs a quadratic approximation of the augmented Lagrangian $L(\cdot,\cdot)$ when solving* (14)*. In other words, unlike our proposed Newton tracking algorithm, ESOM does not linearize the topology-dependent quadratic term $\frac{\alpha}{2}\mathbf{x}^T(\mathbf{I}-\mathbf{W})\mathbf{x}$. As we have indicated earlier, this renders the closed-form solution of the resulting update not implementable in a decentralized manner. Indeed, the primal update of ESOM, which is given by*

$$\begin{aligned}\mathbf{x}^{t+1} = \mathbf{x}^t &- \left(\nabla^2 f(\mathbf{x}) + \alpha(\mathbf{I}-\mathbf{W}) + \epsilon\mathbf{I}\right)^{-1}\\ &\left[\nabla f\left(\mathbf{x}^t\right) + (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^t + \alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t\right],\end{aligned} \tag{22}$$

involves computing the inverse of $\nabla^2 f(\mathbf{x}) + \alpha(\mathbf{I}-\mathbf{W}) + \epsilon\mathbf{I}$. Such a task requires multiple rounds of communication and computation. Although ESOM introduces an inner loop to approximately solve (22)*, it still leads to extra communication and computation costs.*

Remark 2. *The work [33] proposes the Newton-Raphson consensus method to solve problem* (2)*. With proper initialization, it updates x_i^{t+1} on node i as*

$$\begin{aligned}x_i^{t+1} &= (1-\alpha)x_i^t + \alpha\left[H_i^t\right]_\varsigma^{-1} y_i^t,\\ y_i^{t+1} &= \sum_{j\in\mathcal{N}_i} w_{ij}\left(y_j^t + \nabla f_j(x_j^t) - \nabla f_j(x_j^{t-1})\right),\\ H_i^{t+1} &= \sum_{j\in\mathcal{N}_i} w_{ij}\left(H_j^t + \nabla^2 f_j(x_j^t) - \nabla^2 f_j(x_j^{t-1})\right),\end{aligned}$$

where $\alpha \in (0,1]$ is the step size and $[\cdot]_\varsigma$ is a thresholding operator with parameter $\varsigma > 0$ defined in [33]. Compared with Newton tracking, Newton-Raphson consensus requires two rounds of communication in each iteration, one to transmit the gradient trackers $y_i^t \in \mathbb{R}^p$ and another to transmit the Hessian trackers $H_i^t \in \mathbb{R}^{p\times p}$. Note that when p is large, transmitting $p\times p$ matrices over the network is prohibitive due to the high communication cost. In addition, the analysis of Newton-Raphson consensus is different from that in our work.

IV. Convergence Analysis

Since the Newton tracking updates (7) and (8) are equivalent to the primal-dual updates (16) and (15), once we show that the latter exhibits a linear convergence rate, then so does the former. In the analysis, we need the following assumption.

Assumption 2. *The local objective functions $\{f_i\}_{i=1}^n$ are twice differentiable. Moreover, there exist constants $\mu_f, L_f \in (0,+\infty)$ such that*

$$\mu_f I_p \preceq \nabla^2 f_i(x_i) \preceq L_f I_p \tag{23}$$

for all $x_i \in \mathbb{R}^p$ and $i = 1,\ldots,n$.

The lower bound in (23) implies that the local objective functions $\{f_i\}_{i=1}^n$ are strongly convex with parameter $\mu_f > 0$, while the upper bound implies that the local gradients $\{\nabla f_i\}_{i=1}^n$ are Lipschitz continuous with constant $L_f > 0$. Since the Hessian $\nabla^2 f(\mathbf{x})$ of the aggregate objective function f at $\mathbf{x} = [x_1;\ldots;x_n]$ is the block diagonal matrix whose i-th diagonal block is $\nabla^2 f_i(x_i)$, the bounds in (23) also hold for $\nabla^2 f$; i.e.,

$$\mu_f \mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq L_f \mathbf{I}$$

for all $\mathbf{x} \in \mathbb{R}^{np}$. In other words, the aggregate objective function f is also strongly convex with parameter μ_f and its gradient ∇f is Lipschitz continuous with constant L_f.

Our analysis involves the optimal primal-dual pair $(\mathbf{x}^*, \mathbf{v}^*)$ of (13). According to the KKT conditions of (13), we have

$$\nabla f(\mathbf{x}^*) + (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^* = \mathbf{0}, \tag{24}$$

$$(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^* = \mathbf{0}. \tag{25}$$

Lemma 1. *The primal-dual iterates generated by the equivalent Newton tracking updates* (16) *and* (15) *satisfy*

$$\begin{aligned}&\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^*\right) + (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\left(\mathbf{v}^{t+1} - \mathbf{v}^*\right)\\ &\quad + \epsilon\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) + \mathbf{e}^t = \mathbf{0},\end{aligned} \tag{26}$$

where $\mathbf{e}^t$ *is defined as*

$$\mathbf{e}^t \triangleq \nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right) + \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) - \alpha(\mathbf{I} - \mathbf{W})(\mathbf{x}^{t+1} - \mathbf{x}^t).$$

Lemma 1 reveals the relationship of the primal-dual pairs $(\mathbf{x}^t, \mathbf{v}^t)$ and $(\mathbf{x}^{t+1}, \mathbf{v}^{t+1})$ with the optimal primal-dual pair $(\mathbf{x}^*, \mathbf{v}^*)$. It can be proven using arguments similar to those in [31].

Proof. By definition of $\mathbf{e}^t$, (16) can be rewritten as

$$\nabla f\left(\mathbf{x}^{t+1}\right) + (\mathbf{I} - \mathbf{W})^{\frac{1}{2}}\mathbf{v}^t + \alpha(\mathbf{I} - \mathbf{W})\mathbf{x}^{t+1} + \epsilon\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) + \mathbf{e}^t = \mathbf{0}. \tag{27}$$

Combining (24) and (25) with (27), we have

$$\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^*\right) + (\mathbf{I} - \mathbf{W})^{\frac{1}{2}}\left(\mathbf{v}^t - \mathbf{v}^*\right) + \alpha(\mathbf{I} - \mathbf{W})\left(\mathbf{x}^{t+1} - \mathbf{x}^*\right) + \epsilon\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) + \mathbf{e}^t = \mathbf{0}. \tag{28}$$

Now, observe that $\mathbf{v}^t$ in (28) can be further replaced by $\mathbf{v}^{t+1}$. To be specific, substituting (25) into (15) and then regrouping terms, we know that $\mathbf{v}^t$ can be represented as

$$\mathbf{v}^t = \mathbf{v}^{t+1} - \alpha(\mathbf{I} - \mathbf{W})^{\frac{1}{2}}\left(\mathbf{x}^{t+1} - \mathbf{x}^*\right). \tag{29}$$

Substituting (29) into (28) yields (26). □

The term $\mathbf{e}^t$ can be interpreted as the error introduced by approximation at time t. To bound this error, let us introduce the following assumption, which is common in the analysis of second-order methods.

Assumption 3. *The Hessian* $\nabla^2 f$ *of the aggregate objective function* f *is Lipschitz continuous with constant* $L \in (0, +\infty)$*; i.e.,*

$$\left\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\right\| \leq L\|\mathbf{x} - \mathbf{y}\|, \ \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^{np}.$$

In the following lemma, we provide an upper bound on $\|\mathbf{e}^t\|$ in terms of $\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|$. The arguments used in the proof are similar to those in [30].

Lemma 2. *Under Assumptions 2 and 3, the error vectors* $\{\mathbf{e}^t\}_{t\geq 0}$ *associated with the equivalent Newton tracking updates* (16) *and* (15) *satisfy*

$$\left\|\mathbf{e}^t\right\| \leq \left(\rho_t + \alpha\lambda_{\max}(\mathbf{I} - \mathbf{W})\right)\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|, \tag{30}$$

where $\rho_t \triangleq \min\left\{2L_f, \frac{L}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|\right\}$.

Proof. Since $\nabla^2 f(\mathbf{x}) \preceq L_f\mathbf{I}$ for any $\mathbf{x} \in \mathbb{R}^{np}$, we have

$$\begin{aligned}&\left\|\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right\| \\ &\leq \int_0^1 \left\|\nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\right\| ds \\ &\leq L_f\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|\end{aligned}$$

and

$$\|\nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \leq L_f\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|.$$

It follows that

$$\begin{aligned}&\|\nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right) + \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &\leq \|\nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right)\| + \|\nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &\leq 2L_f\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|.\end{aligned} \tag{31}$$

Moreover, since

$$\begin{aligned}&\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right) \\ &= \int_0^1 \nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) ds \\ &= \nabla^2 f(\mathbf{x}^t)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) \\ &\quad + \int_0^1 \left[\nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right) - \nabla^2 f\left(\mathbf{x}^t\right)\right] \left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) ds,\end{aligned}$$

we have

$$\begin{aligned}&\|\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right) - \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &= \left\| \int_0^1 \left[\nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right) - \nabla^2 f\left(\mathbf{x}^t\right)\right] \left(\mathbf{x}^{t+1} - \mathbf{x}^t\right) ds \right\| \\ &\leq \int_0^1 \left\|\nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right) - \nabla^2 f\left(\mathbf{x}^t\right)\right\| \left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\| ds.\end{aligned} \tag{32}$$

By Assumption 3,

$$\left\|\nabla^2 f\left(s\mathbf{x}^{t+1} + (1-s)\mathbf{x}^t\right) - \nabla^2 f\left(\mathbf{x}^t\right)\right\| \leq sL\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|.$$

Substituting this into (32) and computing the integral, we obtain

$$\begin{aligned}&\|\nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right) + \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &\leq \frac{L}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|^2.\end{aligned} \tag{33}$$

It then follows from (31) and (33) that

$$\begin{aligned}&\|\nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right) + \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &\leq \min\left\{2L_f, \frac{L}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|\right\}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\| \\ &= \rho_t\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|,\end{aligned} \tag{34}$$

where $\rho_t \triangleq \min\left\{2L_f, \frac{L}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|\right\}$. Thus, by the triangle inequality,

$$\begin{aligned}\|\mathbf{e}^t\| \leq &\|\nabla f\left(\mathbf{x}^t\right) - \nabla f\left(\mathbf{x}^{t+1}\right) + \nabla^2 f\left(\mathbf{x}^t\right)\left(\mathbf{x}^{t+1} - \mathbf{x}^t\right)\| \\ &+ \|\alpha(\mathbf{I} - \mathbf{W})(\mathbf{x}^{t+1} - \mathbf{x}^t)\| \\ &\leq \left(\rho_t + \alpha\lambda_{\max}(\mathbf{I} - \mathbf{W})\right)\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|,\end{aligned}$$

which completes the proof. □

Lemma 2 shows that the error $\mathbf{e}^t$ introduced by the approximation tends to zero as the sequence of iterates $\{\mathbf{x}^t\}_{t\geq 0}$ approaches the optimal solution $\mathbf{x}^*$.

Given the preliminary results in Lemmas 1 and 2, we are ready to establish the linear convergence of the proposed Newton tracking method. Let $\zeta^t, \zeta^* \in \mathbb{R}^{2np}$ and $\mathbf{G} \in \mathbb{R}^{np\times np}$ be defined as

$$\zeta^t = \begin{bmatrix}\mathbf{x}^t \\ \mathbf{v}^t\end{bmatrix}, \ \zeta^* = \begin{bmatrix}\mathbf{x}^* \\ \mathbf{v}^*\end{bmatrix}, \ \mathbf{G} = \begin{bmatrix}\mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \frac{1}{\alpha}\mathbf{I}\end{bmatrix},$$

where $\mathbf{Q} \triangleq \epsilon\mathbf{I} - \alpha(\mathbf{I}-\mathbf{W})$. Note that $\mathbf{Q}$ is positive definite when $\epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}) > 0$. The following lemma provides a recursion of the primal-dual iterates $\{(\mathbf{x}^t, \mathbf{v}^t)\}_{t\geq 0}$.

Lemma 3. *Under Assumptions 1-3, the primal-dual iterates $\{(\mathbf{x}^t, \mathbf{v}^t)\}_{t\geq 0}$ generated by the equivalent Newton tracking updates* (16) *and* (15) *satisfy*

$$\begin{aligned}&\|\mathbf{x}^* - \mathbf{x}^t\|^2_{\mathbf{Q}} - \|\mathbf{x}^* - \mathbf{x}^{t+1}\|^2_{\mathbf{Q}} \\ &+ \frac{1}{\alpha}\left(\|\mathbf{v}^* - \mathbf{v}^t\|^2 - \|\mathbf{v}^* - \mathbf{v}^{t+1}\|^2\right) \\ \geq &\|\mathbf{x}^t - \mathbf{x}^{t+1}\|^2_{(\mathbf{Q}-\frac{\rho_t^2}{\theta}\mathbf{I})} + \frac{1}{\alpha}\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 \\ &+ (\mu_f - \theta)\left\|\mathbf{x}^* - \mathbf{x}^{t+1}\right\|^2,\end{aligned} \tag{35}$$

where $\theta > 0$ is an arbitrary constant.

Proof. See Appendix. □

Using Lemma 3, we can show that the sequence $\{\|\zeta^t - \zeta^*\|_{\mathbf{G}}\}_{t\geq 0}$ converges to zero at a linear rate.

Theorem 1. *Under Assumptions 1-3, suppose that for all $t \geq 0$, the parameters ϵ and α satisfy $\lambda_{\min}(\mathbf{Q}) = \epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}) \geq \Upsilon + \frac{\rho_t^2}{\mu_f} > \frac{\rho_t^2}{\mu_f}$ for some constant $\Upsilon > 0$. Given any $\beta, \phi > 1$, let*

$$\begin{aligned}\delta_t' = \min\Bigg\{&\frac{\delta_t\mu_f}{(1+\delta_t)\left[\epsilon + \frac{\beta\phi L_f^2}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\right]}, \\ &\frac{\alpha\delta_t^2(\epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}))\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}{\frac{\beta\epsilon^2}{(\beta-1)} + \frac{\beta\phi(\rho_t + \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}))^2}{(\phi-1)}}\Bigg\} > 0,\end{aligned} \tag{36}$$

where

$$\delta_t \triangleq 1 - \frac{\rho_t^2}{\mu_f\lambda_{\min}(\mathbf{Q})} = 1 - \frac{\rho_t^2}{\mu_f(\epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}))} > 0. \tag{37}$$

Then, the primal-dual iterates $\{\zeta^t\}_{t\geq 0}$ generated by the equivalent Newton tracking updates (16) *and* (15) *satisfy*

$$\left\|\zeta^{t+1} - \zeta^*\right\|^2_{\mathbf{G}} \leq \frac{1}{1+\delta_t'}\left\|\zeta^t - \zeta^*\right\|^2_{\mathbf{G}}. \tag{38}$$

Moreover, defining $\underline{\delta'} \triangleq \inf_{t\geq 0}\delta_t'$, we have $\underline{\delta'} > 0$ and thus

$$\left\|\zeta^{t+1} - \zeta^*\right\|^2_{\mathbf{G}} \leq \frac{1}{1+\underline{\delta'}}\left\|\zeta^t - \zeta^*\right\|^2_{\mathbf{G}}. \tag{39}$$

Proof. We first show that $\underline{\delta'} > 0$. Observe that by adjusting $\epsilon, \alpha > 0$ if necessary, we can always find $\Upsilon > 0$ such that

$$\lambda_{\min}(\mathbf{Q}) = \epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}) \geq \Upsilon + \frac{\rho_t^2}{\mu_f} > \frac{\rho_t^2}{\mu_f}.$$

By substituting such $\lambda_{\min}(\mathbf{Q})$ into the definition of δ_t in (37), we have

$$\begin{aligned}\delta_t &= 1 - \frac{\rho_t^2}{\mu_f\lambda_{\min}(\mathbf{Q})} \\ &\geq 1 - \frac{\rho_t^2}{\mu_f(\Upsilon + \frac{\rho_t^2}{\mu_f})} = 1 - \frac{1}{\mu_f(\frac{\Upsilon}{\rho_t^2} + \frac{1}{\mu_f})} \\ &\geq 1 - \frac{1}{\mu_f(\frac{\Upsilon}{4L_f^2} + \frac{1}{\mu_f})} \triangleq \underline{\delta} > 0,\end{aligned}$$

where we use $\rho_t = \min\left\{2L_f, \frac{L}{2}\left\|\mathbf{x}^{t+1} - \mathbf{x}^t\right\|\right\} \leq 2L_f$ in the last inequality. Now, by substituting the bounds $\delta_t \geq \underline{\delta}$, $\lambda_{\min}(\mathbf{Q}) = \epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}) \geq \Upsilon$, and $\rho_t \leq 2L_f$ into the definition of δ_t' in (36), we have

$$\begin{aligned}\delta_t' \geq \min\Bigg\{&\frac{\underline{\delta}\mu_f}{(1+\underline{\delta})\left[\epsilon + \frac{\beta\phi L_f^2}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\right]}, \\ &\frac{\alpha\underline{\delta}^2\Upsilon\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}{\frac{\beta\epsilon^2}{(\beta-1)} + \frac{\beta\phi(2L_f + \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}))^2}{(\phi-1)}}\Bigg\}.\end{aligned}$$

Since the right-hand side of the above inequality does not depend on t and is strictly positive, it follows that $\underline{\delta'} = \inf_{t\geq 0}\delta_t' > 0$, as desired.

Next, we prove (38). From (35), we need to choose $\theta > 0$ such that

$$\begin{cases}\lambda_{\min}(\mathbf{Q}) - \frac{\rho_t^2}{\theta} > 0, \\ \mu_f - \theta > 0.\end{cases}$$

This can be achieved when

$$\delta_t = 1 - \frac{\rho_t^2}{\mu_f\lambda_{\min}(\mathbf{Q})} > 0,$$

which holds if

$$\lambda_{\min}(\mathbf{Q}) = \epsilon - \alpha\lambda_{\max}(\mathbf{I}-\mathbf{W}) > \frac{\rho_t^2}{\mu_f}.$$

In particular, we can choose $\theta = \frac{\mu_f}{1+\delta_t}$ so that $\frac{\rho_t^2}{\theta} = (1-\delta_t^2)\lambda_{\min}(\mathbf{Q})$ and (35) becomes

$$\begin{aligned}&\|\mathbf{x}^* - \mathbf{x}^t\|^2_{\mathbf{Q}} - \|\mathbf{x}^* - \mathbf{x}^{t+1}\|^2_{\mathbf{Q}} \\ &+ \frac{1}{\alpha}\left(\|\mathbf{v}^* - \mathbf{v}^t\|^2 - \|\mathbf{v}^* - \mathbf{v}^{t+1}\|^2\right) \\ \geq &\delta_t^2\lambda_{\min}(\mathbf{Q})\|\mathbf{x}^t - \mathbf{x}^{t+1}\|^2 + \frac{1}{\alpha}\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 \\ &+ \frac{\mu_f\delta_t}{1+\delta_t}\left\|\mathbf{x}^* - \mathbf{x}^{t+1}\right\|^2.\end{aligned} \tag{40}$$

To establish (38), it suffices to show that $\|\zeta^t - \zeta^*\|^2_{\mathbf{G}} - \left\|\zeta^{t+1} - \zeta^*\right\|^2_{\mathbf{G}} \geq \delta_t'\left\|\zeta^{t+1} - \zeta^*\right\|^2_{\mathbf{G}}$. In view of (40), it is enough to show that

$$\begin{aligned}&\frac{\delta_t'}{\alpha}\left\|\mathbf{v}^{t+1} - \mathbf{v}^*\right\|^2 + \delta_t'\left\|\mathbf{x}^{t+1} - \mathbf{x}^*\right\|^2_{\mathbf{Q}} \\ \leq &\delta_t^2\lambda_{\min}(\mathbf{Q})\|\mathbf{x}^t - \mathbf{x}^{t+1}\|^2 + \frac{1}{\alpha}\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 \\ &+ \frac{\mu_f\delta_t}{1+\delta_t}\left\|\mathbf{x}^* - \mathbf{x}^{t+1}\right\|^2.\end{aligned} \tag{41}$$

Towards that end, we first use (26) and the Cauchy-Schwarz inequality to get

$$\begin{aligned}&\left\|\mathbf{v}^{t+1} - \mathbf{v}^*\right\|^2_{\mathbf{I}-\mathbf{W}} \\ \leq &\frac{\beta\epsilon^2}{\beta-1}\|\mathbf{x}^{t+1} - \mathbf{x}^t\|^2 + \beta\phi\|\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^*\right)\|^2 \\ &+ \frac{\beta\phi}{\phi-1}\|\mathbf{e}^t\|^2\end{aligned} \tag{42}$$

for any $\beta, \phi > 1$. By Assumption 2, we have $\|\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^*\right)\|^2 \leq L_f^2\|\mathbf{x}^{t+1} - \mathbf{x}^*\|^2$. Moreover, by (30), we have

$\|\mathbf{e}^t\|^2 \leq \tilde{\rho}_t^2 \left\|\mathbf{x}^{t+1}-\mathbf{x}^t\right\|^2$, where $\tilde{\rho}_t \triangleq \rho_t + \alpha\lambda_{\max}(\mathbf{I}-\mathbf{Z})$. Substituting these inequalities into (42) yields

$$\begin{aligned}&\left\|\mathbf{v}^{t+1}-\mathbf{v}^*\right\|^2_{\mathbf{I}-\mathbf{W}}\\ &\leq\left(\frac{\beta\epsilon^2}{\beta-1}+\frac{\beta\phi\tilde{\rho}_t^2}{\phi-1}\right)\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2+\beta\phi L_f^2\|\mathbf{x}^{t+1}-\mathbf{x}^*\|^2.\end{aligned}$$

Next, considering that $\mathbf{v}^{t+1}$ and $\mathbf{v}^*$ both lie in the column space of $(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}$, we have

$$\begin{aligned}&\left\|\mathbf{v}^{t+1}-\mathbf{v}^*\right\|^2 \leq \frac{1}{\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\\ &\left\{\left(\frac{\beta\epsilon^2}{\beta-1}+\frac{\beta\phi\tilde{\rho}_t^2}{\phi-1}\right)\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2+\beta\phi L_f^2\|\mathbf{x}^{t+1}-\mathbf{x}^*\|^2\right\}.\end{aligned} \tag{43}$$

Here, note that $\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})>0$ because $\mathbf{I}-\mathbf{W}\succeq 0$. In addition, we have

$$\left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|^2_{\mathbf{Q}} \leq \lambda_{\max}(\mathbf{Q})\left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|^2. \tag{44}$$

By substituting (43) and (44) into (41), we see that the following is a sufficient condition for (38) to hold:

$$\begin{aligned}&\lambda_{\max}(\mathbf{Q})\delta_t'\left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|^2+\frac{\delta_t'}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\\ &\left\{\left(\frac{\beta\epsilon^2}{\beta-1}+\frac{\beta\phi\tilde{\rho}_t^2}{\phi-1}\right)\|\mathbf{x}^{t+1}-\mathbf{x}^t\|^2+\beta\phi L_f^2\|\mathbf{x}^{t+1}-\mathbf{x}^*\|^2\right\}\\ \leq&\delta_t^2\lambda_{\min}(\mathbf{Q})\|\mathbf{x}^t-\mathbf{x}^{t+1}\|^2+\frac{1}{\alpha}\|\mathbf{v}^{t+1}-\mathbf{v}^t\|^2\\ &+\frac{\mu_f\delta_t}{1+\delta_t}\left\|\mathbf{x}^*-\mathbf{x}^{t+1}\right\|^2.\end{aligned}$$

After rearranging the terms, the above is equivalent to

$$\begin{aligned}&\left(\frac{\mu_f\delta_t}{1+\delta_t}-\delta_t'\lambda_{\max}(\mathbf{Q})-\frac{\delta_t'\beta\phi L_f^2}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\right)\left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|^2\\ &+\left(\delta_t^2\lambda_{\min}(\mathbf{Q})-\frac{\delta_t{}'\beta\epsilon^2/(\beta-1)}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}-\frac{\delta_t'\beta\phi\tilde{\rho}_t^2/(\phi-1)}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\right)\\ &\left\|\mathbf{x}^{t+1}-\mathbf{x}^t\right\|^2+\frac{1}{\alpha}\left\|\mathbf{v}^{t+1}-\mathbf{v}^t\right\|^2\geq 0.\end{aligned}$$

Clearly, the above inequality holds if the coefficients on the left-hand side are all non-negative. The latter can be guaranteed if

$$\begin{aligned}\delta_t' \leq \min\Bigg\{&\frac{\mu_f\delta_t}{(1+\delta_t)\left[\lambda_{\max}(\mathbf{Q})+\frac{\beta\phi L_f^2}{\alpha\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\right]},\\ &\frac{\alpha\delta_t^2\lambda_{\min}(\mathbf{Q})\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}{\frac{\beta\epsilon^2}{(\beta-1)}+\frac{\beta\phi\tilde{\rho}_t^2}{(\phi-1)}}\Bigg\}.\end{aligned} \tag{45}$$

Since $\mathbf{Q}=\epsilon\mathbf{I}-\alpha(\mathbf{I}-\mathbf{W})$, we have

$$\begin{aligned}\lambda_{\min}(\mathbf{Q})&=\epsilon-\alpha\lambda_{\max}(\mathbf{I}-\mathbf{W})>\frac{\rho_t^2}{\mu_f}>0,\\ \lambda_{\max}(\mathbf{Q})&=\epsilon-\alpha\lambda_{\min}(\mathbf{I}-\mathbf{W})=\epsilon>0.\end{aligned}$$

These, together with the fact that $\tilde{\rho}_t=\rho_t+\alpha\lambda_{\max}(\mathbf{I}-\mathbf{W})$, imply that the choice of δ_t' in (36) satisfies (45), which is sufficient to establish the inequality (38) and hence the linear convergence of the equivalent Newton tracking updates (16) and (15). □

Theorem 1 shows that the sequence $\{\|\zeta^t-\zeta^*\|^2_{\mathbf{G}}\}_{t\geq 0}$ converges to zero linearly, with the factor of linear convergence being $\frac{1}{1+\underline{\delta'}}$. By the definitions of ζ^t and ζ^*, we know that $\|\mathbf{x}^t-\mathbf{x}^*\|^2_{\mathbf{Q}}\leq\|\zeta^t-\zeta^*\|^2_{\mathbf{G}}$. It follows that

$$\left\|\mathbf{x}^t-\mathbf{x}^*\right\|^2_{\mathbf{Q}}\leq\frac{1}{(1+\underline{\delta'})^t}\left\|\zeta^0-\zeta^*\right\|^2_{\mathbf{G}}.$$

In particular, since $\lambda_{\min}(\mathbf{Q})>0$, we conclude that $\mathbf{x}^t$ converges to $\mathbf{x}^*$ linearly.

We require the parameters ϵ and α to satisfy $\epsilon-\alpha\lambda_{\max}(\mathbf{I}-\mathbf{W})>\frac{\rho_t^2}{\mu_f}$ for all $t\geq 0$. Since $\lambda_{\max}(\mathbf{I}-\mathbf{W})<2$ and $\rho_t=\min\left\{2L_f,\frac{L}{2}\left\|\mathbf{x}^{t+1}-\mathbf{x}^t\right\|\right\}\leq 2L_f$, a sufficient condition on the step sizes is $\epsilon-2\alpha>\frac{4L_f^2}{\mu_f}$. There are several works on how to estimate the global parameters L_f and μ_f in a decentralized manner [41], [42]. Even if the estimated L_f and μ_f are inaccurate, a sufficiently large ϵ and a sufficiently small α will satisfy the condition $\epsilon-2\alpha>\frac{4L_f^2}{\mu_f}$.

Now, let us investigate the behavior of the equivalent Newton tracking updates (16) and (15) when $\mathbf{x}^t$ is close to $\mathbf{x}^*$ and hence ρ_t is small. Define the condition numbers of the objective function and the graph as

$$\kappa_f=\frac{L_f}{\mu_f},\quad \kappa_g=\frac{\lambda_{\max}(\mathbf{I}-\mathbf{W})}{\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})},$$

respectively. The following corollary reveals how these two quantities affect the asymptotic convergence rate of Newton tracking.

Corollary 1. *Under Assumptions 1-3, suppose that we take*

$$\begin{aligned}\alpha&=\frac{\sqrt{\kappa_g}L_f}{\lambda_{\max}(\mathbf{I}-\mathbf{W})},\\ \epsilon&=2\left(\alpha\lambda_{\max}(\mathbf{I}-\mathbf{W})+\frac{2L_f^2}{\mu_f}\right)=2\left(\sqrt{\kappa_g}L_f+\frac{2L_f^2}{\mu_f}\right).\end{aligned} \tag{46}$$

Then, we have

$$\delta_t'=\Omega\left(\min\left\{\frac{1}{\kappa_f\sqrt{\kappa_g}+\kappa_f^2},\frac{1}{\frac{\kappa_g^{3/2}}{\kappa_f}+\kappa_f\sqrt{\kappa_g}}\right\}\right).$$

Consequently, when $\mathbf{x}^t$ is close to $\mathbf{x}^*$, Newton tracking requires $O\left(\max\left\{\kappa_f\sqrt{\kappa_g}+\kappa_f^2,\frac{\kappa_g^{3/2}}{\kappa_f}+\kappa_f\sqrt{\kappa_g}\right\}\log\frac{1}{\Delta}\right)$ iterations to reach a Δ-optimal solution (i.e., a solution that is within Δ of $\mathbf{x}^*$).

Proof. For simplicity, let us write $\lambda_{\max}=\lambda_{\max}(\mathbf{I}-\mathbf{W})$ and $\hat{\lambda}_{\min}=\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})$ in the proof. Since $\rho_t\leq 2L_f$ for all $t\geq 0$, with the parameters given by (46), we have

$$\lambda_{\min}(\mathbf{Q})=\epsilon-\alpha\lambda_{\max}=\sqrt{\kappa_g}L_f+\frac{4L_f^2}{\mu_f}>\frac{\rho_t^2}{\mu_f},$$

for all $t\geq 0$. Therefore, Theorem 1 holds. Here, we choose $\Upsilon=\sqrt{\kappa_g}L_f$ in Theorem 1 so that $\delta_t\geq\underline{\delta}$ and $\delta_t'\geq\underline{\delta'}$ for all t. In the following, we will further refine δ_t and δ_t' when $\mathbf{x}^t$ is close to $\mathbf{x}^*$.

Since $\mathbf{x}^t$ converges to $\mathbf{x}^*$ by Theorem 1, there exists a $T \geq 0$ such that $\rho_t \leq \sqrt{\mu_f \lambda_{\min}(\mathbf{Q})/2}$ and hence $\delta_t \geq 1/2$ for all $t \geq T$. Now, recall from (36) that δ_t' is the minimum of two terms. When $t \geq T$, the first term reduces to

$$\begin{aligned}
&\frac{\delta_t \mu_f}{(1+\delta_t)\left[\epsilon + \frac{\beta\phi L_f^2}{\alpha\hat{\lambda}_{\min}}\right]} \qquad (47)\\
=&\frac{\delta_t \mu_f}{(1+\delta_t)\left[2\left(\sqrt{\kappa_g}L_f + \frac{2L_f^2}{\mu_f}\right) + \frac{\beta\phi L_f \lambda_{\max}}{\sqrt{\kappa_g}\hat{\lambda}_{\min}}\right]}\\
=&\frac{\delta_t}{(1+\delta_t)\left[2\left(\sqrt{\kappa_g}\kappa_f + \frac{2L_f^2}{\mu_f^2}\right) + \beta\phi\kappa_f\sqrt{\kappa_g}\right]}\\
=&\Omega\left(\frac{1}{\kappa_f\sqrt{\kappa_g}+\kappa_f^2}\right),
\end{aligned}$$

while the second term reduces to

$$\begin{aligned}
&\frac{\alpha\delta_t^2(\epsilon-\alpha\lambda_{\max})\hat{\lambda}_{\min}}{\frac{\beta\epsilon^2}{(\beta-1)} + \frac{\beta\phi(\rho_t+\alpha\lambda_{\max})^2}{(\phi-1)}} \qquad (48)\\
=&\frac{\alpha\delta_t^2\left(\alpha\lambda_{\max}+\frac{4L_f^2}{\mu_f}\right)\hat{\lambda}_{\min}}{\frac{4\beta}{\beta-1}\left(\alpha\lambda_{\max}+\frac{2L_f^2}{\mu_f}\right)^2 + \frac{\beta\phi(\rho_t+\alpha\lambda_{\max})^2}{(\phi-1)}}\\
=&\frac{\delta_t^2 + \frac{4\delta_t^2 L_f^2}{\mu_f\alpha\lambda_{\max}}}{\frac{4\beta}{\beta-1}\left(\sqrt{\kappa_g}+\frac{2L_f^2/\alpha\mu_f}{\sqrt{\lambda_{\max}\hat{\lambda}_{\min}}}\right)^2 + \frac{\beta\phi}{\phi-1}\left(\frac{\rho_t/\alpha}{\sqrt{\lambda_{\max}\hat{\lambda}_{\min}}}+\sqrt{\kappa_g}\right)^2}\\
\geq&\frac{\delta_t^2 + \frac{4\delta_t^2\kappa_f}{\sqrt{\kappa_g}}}{\frac{4\beta}{\beta-1}\left(\sqrt{\kappa_g}+2\kappa_f\right)^2 + \frac{\beta\phi}{\phi-1}\left(2+\sqrt{\kappa_g}\right)^2}\\
=&\Omega\left(\frac{\kappa_f/\sqrt{\kappa_g}}{(\sqrt{\kappa_g}+\kappa_f)^2+\kappa_g}\right)\\
=&\Omega\left(\frac{1}{\frac{\kappa_g^{3/2}}{\kappa_f}+\kappa_f\sqrt{\kappa_g}}\right).
\end{aligned}$$

With (47) and (48), we complete the proof. □

Corollary 1 shows that when $\mathbf{x}^t$ is close to $\mathbf{x}^*$, the condition numbers of the objective function and the graph determine the asymptotic convergence rate of our proposed Newton tracking algorithm. In particular, when $\kappa_f \approx \sqrt{\kappa_g}$, the iteration complexity of finding an Δ-optimal solution is $O(\kappa_f^2 \log\frac{1}{\Delta})$. Convergence rates of Newton tracking and several existing algorithms are compared in Table I.

V. Numerical Experiments

In this section, we apply our proposed Newton tracking algorithm to solve a decentralized logistic regression problem of the form

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

where node i has access to the training samples $(\mathbf{o}_{ij}, \mathbf{p}_{ij}) \in \mathbb{R}^p \times \{-1,+1\}$; $j = 1,\ldots,m_i$. We add a regularization term $\frac{\rho}{2}\|x\|^2$ with $\rho > 0$ to the loss function to avoid overfitting. In our numerical experiments, we randomly generate the elements in $\mathbf{o}_{ij}$ according to the normal distribution and the elements in $\mathbf{p}_{ij}$ according to the uniform distribution. Also, we randomly generate $\frac{\tau n(n-1)}{2}$ undirected edges for the network of n nodes, where $\tau \in (0,1]$ is the connectivity ratio and is chosen to ensure that the network is connected.

To evaluate the performance of the compared algorithms, the optimal logistic classifier x^* is pre-computed through centralized gradient descent. We use relative error as the performance metric, which is defined as $\|\mathbf{x}^t - \mathbf{x}^*\| / \left\|\mathbf{x}^0 - \mathbf{x}^*\right\|$. We conduct the experiments with Matlab R2016b, running on a laptop with Intel(R) Core(TM) i7 CPU@1.80GHz, 16.0 GB of RAM, and Windows 10 operating system.

A. Comparison with Second-Order Methods

We compare Newton tracking with other second-order algorithms in the literature, including NN-K [26], ESOM-K [31], and DQM [30]. In every iteration of NN-K and ESOM-K, the nodes need to execute a $(K+1)$-round inner loop to approximately compute the inverse of a topology-dependent matrix of the forms $\alpha\nabla^2 f(\mathbf{x}) + (\mathbf{I}-\mathbf{W})$ and $\nabla^2 f(\mathbf{x}) + \alpha(\mathbf{I}-\mathbf{W}) + \epsilon\mathbf{I}$, respectively.

We set the number of nodes as $n = 10$ and the connectivity ratio as $\tau = 0.5$. Each node holds 12 samples; i.e., $m_i = 12$ for all i. The dimension of the sample vectors $\{\mathbf{o}_{ij}\}$ is $p = 8$. We set the regularization parameter $\rho = 0.001$.

All the algorithms use hand-optimized step sizes. The step size of DQM is set to $\alpha = 0.3$. The parameters of ESOM-0 (ESOM-1; ESOM-2) are set to $\alpha = 3(3.4; 5.5)$ and $\epsilon = 0.1(0.1; 0.1)$. For NN-K, a smaller step size improves accuracy but leads to slower convergence, while a larger step size accelerates the convergence at the cost of lower accuracy. Therefore, we use the step sizes $\alpha = 0.001$, $\alpha = 0.008$, and $\alpha = 0.02$ for NN-0, NN-1, and NN-2, respectively. For Newton tracking, we set $\alpha = 3.9$ and $\epsilon = 3.6$.

Fig. 1 illustrates the relative error versus the number of iterations. Observe that NN-K only converges to a neighborhood of the optimal solution. Among the exact decentralized algorithms, except for ESOM-2, the proposed Newton tracking algorithm has the best performance compared with the other algorithms and converges linearly, which validates the theoretical result in Theorem 1.

Newton tracking and DQM require one round of communication per iteration, while NN-K and ESOM-K require $K+1$

[1]In this table, $\kappa_f = \frac{L_f}{\mu_f}$ and $\kappa_g = \frac{\lambda_{\max}(\mathbf{I}-\mathbf{W})}{\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}$ are the condition numbers of f_i and the graph, respectively; α and ϵ are the step sizes; σ is the second largest absolute eigenvalue of W, i.e., $\sigma = \lambda_{\max}(W - \frac{1}{n}1_n 1_n^T)$.

[2]Here, $\mathcal{L}_u$ and $\mathcal{L}_o$ are the unoriented and oriented Laplacian, respectively, which are defined in [16]. The refined rate is obtained when $\alpha = \frac{L_f\kappa_f}{\lambda_{\min}(\mathcal{L}_u)}$ and $\epsilon = L_f\kappa_f$.

[3]Here, we set $\tilde{W}$ in [17] as $\tilde{W} = \frac{I+W}{2}$ and $\alpha = \frac{0.5\hat{\lambda}_{\min}(\tilde{W})}{L_f\kappa_f}$.

[4]Here, the refined rate is obtained when $\mathbf{x}^t \to \mathbf{x}^*$ and $\alpha = \frac{5L_f\kappa_f}{(\lambda_{\min}(\mathcal{L}_u))^2}$.

[5]Here, $\Gamma_t = \min\left\{2L_f, \frac{L}{2}\|\mathbf{x}^{t+1}-\mathbf{x}^t\|\right\} + (L_f+\epsilon+2\alpha(1-\omega))\rho^{K+1}$, where $\omega = \min_i W_{ii}$, K is the number of inner-loop iterations and $\rho \in (0,1)$ is some constant defined in [31]. The refined rate is obtained when $\alpha = \frac{\mu_f}{20(1-\omega)}$, $\epsilon = 3(\mu_f + L_f)\kappa_f$, and $\Gamma_t \to 0$, meaning that $\mathbf{x}^t \to \mathbf{x}^*$ and $K \to \infty$.

[6]Here, the refined rate is obtained when $\mathbf{x}^t \to \mathbf{x}^*$; see Corollary 1 for details.

TABLE I
CONVERGENCE RATES OF DECENTRALIZED CONSENSUS OPTIMIZATION ALGORITHMS TO SOLVE (2)

Algorithm	Step size	Rate [1]
DLM [16]	$\alpha\lambda_{\min}(\mathcal{L}_u)+\epsilon>\frac{L_f\kappa_f}{2}$	$O\left(\max\left\{\frac{\kappa_f^2\lambda_{\max}(\mathcal{L}_u)}{\lambda_{\min}(\mathcal{L}_u)}+\frac{\lambda_{\min}(\mathcal{L}_u)}{\tilde{\lambda}_{\min}(\mathcal{L}_o)},\frac{(\lambda_{\max}(\mathcal{L}_u))^2}{\lambda_{\min}(\mathcal{L}_u)\tilde{\lambda}_{\min}(\mathcal{L}_o)}+\frac{\lambda_{\min}(\mathcal{L}_u)}{\kappa_f^2\tilde{\lambda}_{\min}(\mathcal{L}_o)}\right\}\log(\frac{1}{\Delta})\right)$ [2]
EXTRA [17]	$\alpha<\frac{2\lambda_{\min}(\tilde{W})}{L_f\kappa_f}$	$O\left(\frac{\kappa_f^2}{1-\sigma}\log(\frac{1}{\Delta})\right)$ [3]
gradient tracking [21]	$\alpha=\frac{(1-\sigma)^2}{36L_f\kappa_f}$	$O\left(\frac{\kappa_f^2}{(1-\sigma)^2}\log(\frac{1}{\Delta})\right)$
DQM [30]	$\alpha>\frac{4L_f\kappa_f}{(\lambda_{\min}(\mathcal{L}_u))^2}$	$O\left(\max\left\{\left(\frac{\lambda_{\max}(\mathcal{L}_u)}{\tilde{\lambda}_{\min}(\mathcal{L}_o)}\right)^2,\kappa_f^2\left(\frac{\lambda_{\max}(\mathcal{L}_u)}{\lambda_{\min}(\mathcal{L}_u)}\right)^2+\mu_f\left(\frac{\lambda_{\min}(\mathcal{L}_u)}{\tilde{\lambda}_{\min}(\mathcal{L}_o)}\right)^2\right\}\log(\frac{1}{\Delta})\right)$ [4]
ESOM [31]	$\epsilon>\frac{\mu_f+L_f}{2\mu_f L_f}\left(2L_f+2\alpha(1-\omega)\kappa_f\right)^2$	$O\left(\frac{\kappa_f^2}{\tilde{\lambda}_{\min}(\mathbf{I}-\mathbf{W})}\log(\frac{1}{\Delta})\right)$ [5]
Newton tracking	$\epsilon-\alpha\lambda_{\max}(\mathbf{I}-\mathbf{W})>\frac{\rho_t^2}{\mu_f}$	$O\left(\max\left\{\kappa_f\sqrt{\kappa_g}+\kappa_f^2,\frac{\kappa_g^{3/2}}{\kappa_f}+\kappa_f\sqrt{\kappa_g}\right\}\log\frac{1}{\Delta}\right)$ [6]

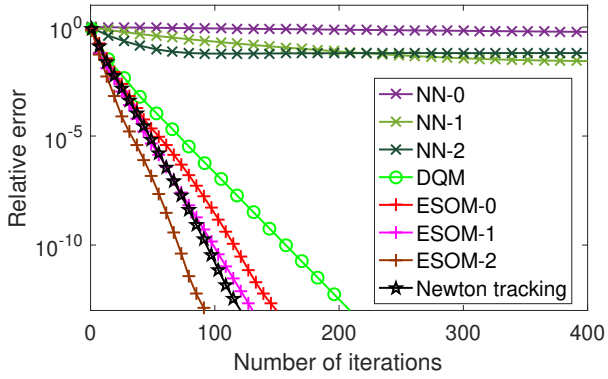


Fig. 1. Relative errors of Newton tracking, DQM, NN-K, and ESOM-K versus number of iterations.

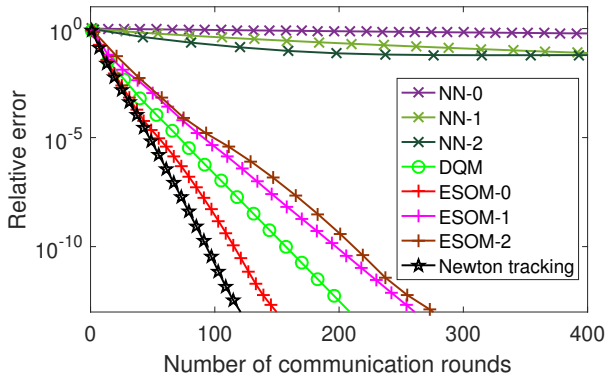


Fig. 2. Relative errors of Newton tracking, NN-K, ESOM-K, and DQM versus rounds of communications.

rounds. Fig. 2 illustrates the relative error versus the number of communication rounds. Observe that although ESOM-1 and ESOM-2 perform well as depicted in Fig. 1, they become worse in Fig. 2 because more rounds of communication are required in each iteration. In terms of communication cost, the proposed Newton tracking algorithm is the best.

B. Comparison with First-Order Methods

We compare Newton tracking with several first-order algorithms, including gradient tracking [21], EXTRA [17], and DLM [16]. Their equivalent updates are given as follows:

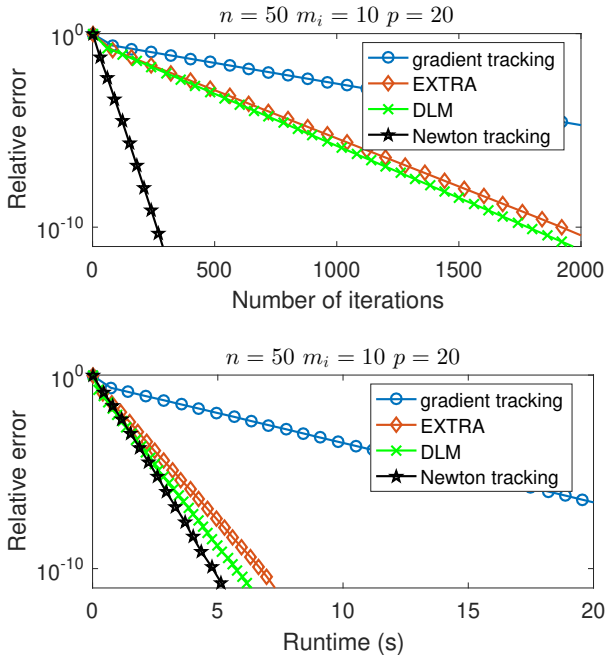


Fig. 3. Relative errors of Newton tracking, gradient tracking, EXTRA and DLM when $n = 50$, $m_i = 10$ and $p = 20$.

- Gradient tracking:

$$\mathbf{x}^{t+2} = 2\mathbf{W}\mathbf{x}^{t+1} - \mathbf{W}^2\mathbf{x}^t - \alpha\left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right].$$

- EXTRA:

$$\mathbf{x}^{t+2} = (\mathbf{I}+\mathbf{W})\mathbf{x}^{t+1} - (\mathbf{I}+\mathbf{W})\mathbf{x}^t/2 - \alpha\left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right].$$

- DLM:

$$\mathbf{x}^{t+2} = (\mathbf{I}-\alpha D L_o)\left(2\mathbf{x}^{t+1}-\mathbf{x}^t\right) - D\left[\nabla f\left(\mathbf{x}^{t+1}\right) - \nabla f\left(\mathbf{x}^t\right)\right].$$

Here, $D = \text{diag}\{1/(2\alpha d_i+\epsilon)\}$, d_i is the degree of node i, and L_o is the oriented Laplacian defined in [16].

We consider two larger networks, which consist of $n = 50$ (100) nodes. The number of samples on each node is $m_i = 10$ (10) for all i and the dimension of the sample vectors

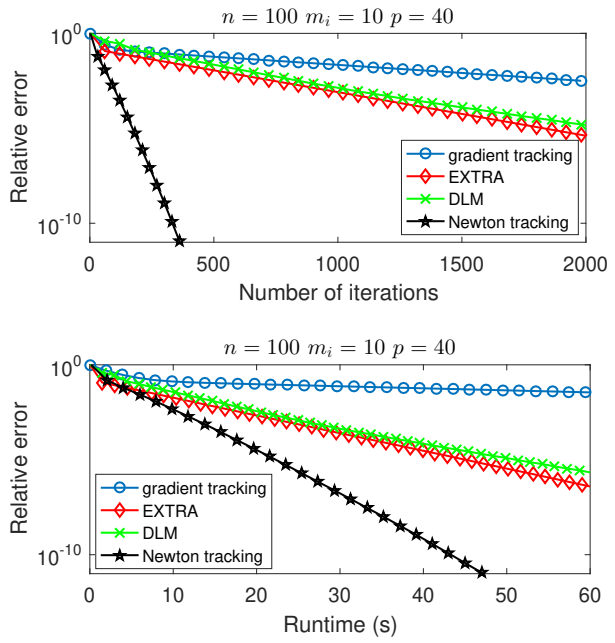


Fig. 4. Relative errors of Newton tracking, gradient tracking, EXTRA, and DLM when $n = 100$, $m_i = 10$, and $p = 40$.

as $p = 20$ (40). We run all the algorithms with fixed hand-optimized step sizes. The step sizes of gradient tracking and EXTRA are set to $\alpha = 0.16$ (0.6) and $\alpha = 0.07$ (1.6), respectively. The parameters of DLM are set to $\alpha = 0.04$ (0.006) and $\epsilon = 4$ (0.1). For Newton tracking, the parameters are set to $\alpha = 1.1$ (0.08) and $\epsilon = 1.2$ (0.08). All other settings are the same as those in Section V-A.

Fig. 3 illustrates the relative error versus the number of iterations and the runtime, respectively, on the $n = 50$ network. Observe that the proposed Newton tracking outperforms the first-order algorithms in terms of both the number of iterations and runtime. Although Newton tracking computes the inverse of the estimated Hessian $\nabla^2 f_i(x_i) + \epsilon I_p \in \mathbb{R}^{p\times p}$ in each iteration, it calls for a relatively smaller number of iterations compared with the first-order algorithms and hence leads to a shorter runtime. Similar results hold for the $n = 100$ network; see Fig. 4. However, when the dimension p is very large, Newton tracking becomes less efficient than first-order methods because computing the inverse of the regularized Hessian $\nabla^2 f_i(x_i) + \epsilon I_p \in \mathbb{R}^{p\times p}$ is time-consuming.

C. Effect of Network Topology

Now, let us investigate the performance of Newton tracking and ESOM-1 in four different topologies – the line graph, the cycle graph, random graphs with $\tau = \{0.3, 0.5, 0.7\}$, and the complete graph. The parameters of ESOM-1 are set to $\alpha = 6$ $(2.9, 3.8, 3.4, 3.2, 2.9)$, $\epsilon = 3$ $(0.1, 0.05, 0.1, 0.1, 0.1)$, and the parameters of Newton tracking are set to $\alpha = 5.9$ $(2.4, 2.7, 3.9, 3.1, 2.9)$, $\epsilon = 5.9$ $(2.4, 2.4, 3.6, 2.7, 2.6)$. All other settings are the same as those in Section V-A.

Figs. 5 and 6 illustrate the relative error versus the number of communication rounds. Newton tracking outperforms ESOM-1 in all the topologies. Observe that Newton tracking

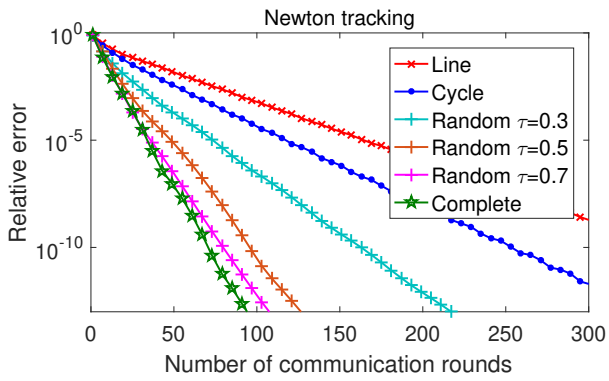


Fig. 5. Relative errors of Newton tracking versus rounds of communications for line graph, cycle graph, random graphs with $\tau = \{0.3, 0.5, 0.7\}$, and complete graph.

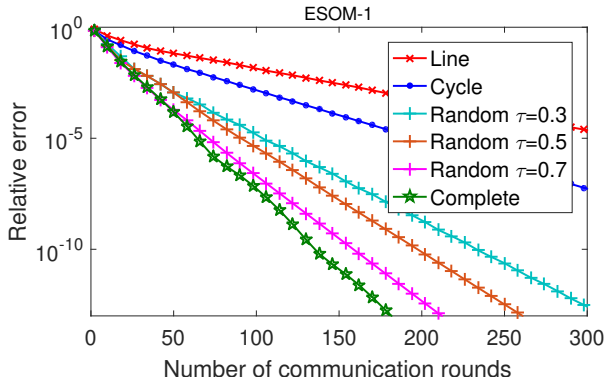


Fig. 6. Relative errors of ESOM-1 versus rounds of communications for line graph, cycle graph, random graphs with $\tau = \{0.3, 0.5, 0.7\}$, and complete graph.

has linear convergence in all types of graphs. Among them, the complete graph yields the fastest speed. This observation confirms our theoretical analysis on the convergence speed. For the line graph, the cycle graph, random graphs with $\tau = \{0.3, 0.5, 0.7\}$, and the complete graph, we have $\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W}) = \{0.03, 0.12, 0.17, 0.34, 0.43, 1.00\}$ and $\lambda_{\max}(\mathbf{I}-\mathbf{W}) = \{1.30, 1.33, 1.16, 1.15, 1.10, 1.00\}$, respectively. According to our theoretical analysis, the complete graph with the largest $\hat{\lambda}_{\min}(\mathbf{I}-\mathbf{W})$ and the smallest $\lambda_{\max}(\mathbf{I}-\mathbf{W})$ has the largest δ_t', hence the fastest convergence speed.

VI. Conclusions

This paper proposed a novel Newton tracking algorithm to solve the decentralized consensus optimization problem. Each node updates its local variable along a modified local Newton direction, which is calculated using neighboring and historical information. Newton tracking employs a fixed step size and yet provably converges to an exact optimal solution. The connections between Newton tracking and several existing methods, including gradient tracking and second-order algorithms, were investigated. We proved that the proposed algorithm converges at a linear rate under the strongly convex assumption. Our numerical results demonstrated the efficacy of Newton tracking and its superiority over existing algorithms such as gradient tracking, NN, ESOM, and DQM.

APPENDIX

Proof. The proof of Lemma 3 has two steps.

Step 1. By reorganizing (16), we get

$$\epsilon(\mathbf{x}^t-\mathbf{x}^{t+1})+\nabla^2 f(\mathbf{x}^t)\left(\mathbf{x}^t-\mathbf{x}^{t+1}\right) - \left[\nabla f\left(\mathbf{x}^t\right)+(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^t+\alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t\right]=\mathbf{0},$$

which implies that

$$\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \epsilon(\mathbf{x}^t-\mathbf{x}^{t+1})+\nabla^2 f(\mathbf{x}^t)\left(\mathbf{x}^t-\mathbf{x}^{t+1}\right) - \left[\nabla f\left(\mathbf{x}^t\right)+(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^t+\alpha(\mathbf{I}-\mathbf{W})\mathbf{x}^t\right]\right\rangle=0. \tag{49}$$

Substituting the dual update $\mathbf{v}^t=\mathbf{v}^{t+1}-\alpha(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}$ and regrouping the terms, we can rewrite (49) as

$$\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \underbrace{(\epsilon\mathbf{I}-\alpha(\mathbf{I}-\mathbf{W}))}_{\triangleq\mathbf{Q}}(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle \tag{50}$$
$$-\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^t\right)\right\rangle-\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^{t+1}\right\rangle + \left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle=0.$$

For the first term on the left-hand side of (50), we have

$$\begin{aligned}&\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \mathbf{Q}(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle \\ =&\frac{1}{2}\left(\|\mathbf{x}^*-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2+\|\mathbf{x}^t-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2-\|\mathbf{x}^*-\mathbf{x}^t\|_{\mathbf{Q}}^2\right).\end{aligned} \tag{51}$$

For the second term on the left-hand side of (50), we use the μ_f-strong convexity of f to get

$$\begin{aligned}&\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^t\right)\right\rangle \\ =&\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^{t+1}\right)\right\rangle \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^t\right)-\nabla f\left(\mathbf{x}^{t+1}\right)\right\rangle \\ \le& f(\mathbf{x}^*)-f\left(\mathbf{x}^{t+1}\right)-\frac{\mu_f}{2}\left\|\mathbf{x}^*-\mathbf{x}^{t+1}\right\|^2 \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^t\right)-\nabla f\left(\mathbf{x}^{t+1}\right)\right\rangle.\end{aligned} \tag{52}$$

Substituting (52) and (51) into (50), we get

$$\begin{aligned}&\frac{1}{2}\left(\|\mathbf{x}^*-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2+\|\mathbf{x}^t-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2-\|\mathbf{x}^*-\mathbf{x}^t\|_{\mathbf{Q}}^2\right) \\ &-f(\mathbf{x}^*)+f\left(\mathbf{x}^{t+1}\right)+\frac{\mu_f}{2}\left\|\mathbf{x}^*-\mathbf{x}^{t+1}\right\|^2 \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^{t+1}\right)-\nabla f\left(\mathbf{x}^t\right)\right\rangle \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle \\ &-\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^{t+1}\right\rangle\le 0.\end{aligned}$$

Upon rearranging the terms above, we obtain

$$\begin{aligned}&\underbrace{f(\mathbf{x}^*)-f\left(\mathbf{x}^{t+1}\right)}_{(i)}+\underbrace{\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^{t+1}\right\rangle}_{(ii)} \\ &-\frac{1}{2}\left(\|\mathbf{x}^*-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2-\|\mathbf{x}^*-\mathbf{x}^t\|_{\mathbf{Q}}^2\right) \\ \ge&\frac{1}{2}\|\mathbf{x}^t-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2+\frac{\mu_f}{2}\left\|\mathbf{x}^*-\mathbf{x}^{t+1}\right\|^2 \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^{t+1}\right)-\nabla f\left(\mathbf{x}^t\right)\right\rangle \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle.\end{aligned} \tag{53}$$

Step 2. According to the dual update (15), we have $\mathbf{v}^{t+1}=\mathbf{v}^t+\alpha(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}$. Consequently,

$$\begin{aligned}&\left\langle \mathbf{v}^*-\mathbf{v}^{t+1}, -(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}\right\rangle \\ =&\left\langle \mathbf{v}^*-\mathbf{v}^{t+1}, \frac{\mathbf{v}^t-\mathbf{v}^{t+1}}{\alpha}\right\rangle \\ =&\frac{1}{2\alpha}\left(\|\mathbf{v}^{t+1}-\mathbf{v}^t\|^2-\|\mathbf{v}^*-\mathbf{v}^t\|^2+\|\mathbf{v}^*-\mathbf{v}^{t+1}\|^2\right).\end{aligned}$$

Rearranging the terms, we have

$$\begin{aligned}&\underbrace{\left\langle \mathbf{v}^*, -(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}\right\rangle}_{(i')}+\underbrace{\left\langle \mathbf{v}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}\right\rangle}_{(ii')} \\ &+\frac{1}{2\alpha}\left(\|\mathbf{v}^*-\mathbf{v}^t\|^2-\|\mathbf{v}^*-\mathbf{v}^{t+1}\|^2\right) \\ =&\frac{1}{2\alpha}\|\mathbf{v}^{t+1}-\mathbf{v}^t\|^2.\end{aligned} \tag{54}$$

Now, we utilize the consensus condition $(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^*=\mathbf{0}$ to sum up (53) and (54). Upon adding (i) and (i'), we have

$$\begin{aligned}&f(\mathbf{x}^*)-f(\mathbf{x}^{t+1})+\left\langle \mathbf{v}^*, -(\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}\right\rangle \\ =&\hat{L}(\mathbf{x}^*,\mathbf{v}^*)-\hat{L}(\mathbf{x}^{t+1},\mathbf{v}^*)\le 0,\end{aligned}$$

where $\hat{L}(\mathbf{x},\mathbf{v})=f(\mathbf{x})+\langle \mathbf{v}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}\rangle$ is the Lagrangian of (13) and the inequality holds because $(\mathbf{x}^*,\mathbf{v}^*)$ is the saddle point of $\hat{L}(\cdot,\cdot)$. On the other hand, adding (ii) and (ii') yields

$$\begin{aligned}&\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^{t+1}\right\rangle+\left\langle \mathbf{v}^{t+1}, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{x}^{t+1}\right\rangle \\ =&\left\langle \mathbf{x}^*, (\mathbf{I}-\mathbf{W})^{\frac{1}{2}}\mathbf{v}^{t+1}\right\rangle=0.\end{aligned}$$

Hence, by adding (53) and (54), we obtain

$$\begin{aligned}&\frac{1}{2}\left(\|\mathbf{x}^*-\mathbf{x}^t\|_{\mathbf{Q}}^2-\|\mathbf{x}^*-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2\right) \\ &+\frac{1}{2\alpha}\left(\|\mathbf{v}^*-\mathbf{v}^t\|^2-\|\mathbf{v}^*-\mathbf{v}^{t+1}\|^2\right) \\ \ge&\frac{1}{2}\|\mathbf{x}^t-\mathbf{x}^{t+1}\|_{\mathbf{Q}}^2+\frac{1}{2\alpha}\|\mathbf{v}^{t+1}-\mathbf{v}^t\|^2+\frac{\mu_f}{2}\left\|\mathbf{x}^*-\mathbf{x}^{t+1}\right\|^2 \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^{t+1}\right)-\nabla f\left(\mathbf{x}^t\right)\right\rangle \\ &+\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle.\end{aligned} \tag{55}$$

To proceed, we observe that for any $\theta>0$,

$$\begin{aligned}&\left\langle \mathbf{x}^*-\mathbf{x}^{t+1}, \nabla f\left(\mathbf{x}^{t+1}\right)-\nabla f\left(\mathbf{x}^t\right)+\nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\right\rangle \\ \ge&-\frac{1}{\theta}\|\nabla f\left(\mathbf{x}^{t+1}\right)-\nabla f\left(\mathbf{x}^t\right)+\nabla^2 f(\mathbf{x}^t)(\mathbf{x}^t-\mathbf{x}^{t+1})\|^2 \\ &-\theta\|\mathbf{x}^*-\mathbf{x}^{t+1}\|^2 \\ \ge&-\theta\|\mathbf{x}^*-\mathbf{x}^{t+1}\|^2-\frac{\rho_t^2}{\theta}\|\mathbf{x}^t-\mathbf{x}^{t+1}\|^2,\end{aligned} \tag{56}$$

where the first inequality follows from the Cauchy-Schwarz inequality and the second follows from (34). Substituting (56)

into (55), we obtain

$$
\begin{aligned}
&\|\mathbf{x}^* - \mathbf{x}^t\|_{\mathbf{Q}}^2 - \|\mathbf{x}^* - \mathbf{x}^{t+1}\|_{\mathbf{Q}}^2 \qquad (57)\\
&+ \frac{1}{\alpha}\left(\|\mathbf{v}^* - \mathbf{v}^t\|^2 - \|\mathbf{v}^* - \mathbf{v}^{t+1}\|^2\right)\\
\geq&\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_{\mathbf{Q}}^2 + \frac{1}{\alpha}\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2 + \mu_f \left\|\mathbf{x}^* - \mathbf{x}^{t+1}\right\|^2\\
&- \theta\|\mathbf{x}^* - \mathbf{x}^{t+1}\|^2 - \frac{\rho_t^2}{\theta}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|^2\\
=&\|\mathbf{x}^t - \mathbf{x}^{t+1}\|^2_{(\mathbf{Q} - \frac{\rho_t^2}{\theta}\mathbf{I})} + \frac{1}{\alpha}\|\mathbf{v}^{t+1} - \mathbf{v}^t\|^2\\
&+ (\mu_f - \theta)\left\|\mathbf{x}^* - \mathbf{x}^{t+1}\right\|^2.
\end{aligned}
$$

which completes the proof. □

REFERENCES

[1] S. Pu, W. Shi, J. Xu, and A. Nedić, "A push-pull gradient method for distributed optimization in networks," in *IEEE Conference on Decision and Control*, 2018, pp. 3385–3390.

[2] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. Leung, "Decentralized resource allocation for video transcoding and delivery in blockchain-based system with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11 169–11 185, 2019.

[3] D.-T. Ta, K. Khawam, S. Lahoud, C. Adjih, and S. Martin, "LoRa-MAB: A flexible simulator for decentralized learning resource allocation in IoT networks," in *IFIP Wireless and Mobile Networking Conference*, 2019, pp. 55–62.

[4] E. Dall'Anese, H. Zhu, and G. B. Giannakis, "Distributed optimal power flow for smart microgrids," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1464–1475, 2013.

[5] H. J. Liu, W. Shi, and H. Zhu, "Hybrid voltage control in distribution networks under limited communication rates," *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 2416–2427, 2019.

[6] A. Lalitha, S. Shekhar, T. Javidi, and F. Koushanfar, "Fully decentralized federated learning," in *Advances in Neural Information Processing Systems Workshop on Bayesian Deep Learning*, 2018.

[7] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[8] Y. Zhao, J. Zhao, L. Jiang, R. Tan, and D. Niyato, "Mobile edge computing, blockchain and reputation-based crowdsourcing IoT federated learning: A secure, decentralized and privacy-preserving system," *arXiv preprint arXiv:1906.10893*, 2019.

[9] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent," in *Advances in Neural Information Processing Systems*, 2017.

[10] A. Koppel, S. Paternain, C. Richard, and A. Ribeiro, "Decentralized online learning with kernels," *IEEE Transactions on Signal Processing*, vol. 66, no. 12, pp. 3240–3255, 2018.

[11] D. Lee, N. He, P. Kamalaruban, and V. Cevher, "Optimization for reinforcement learning: From a single agent to cooperative agents," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 123–135, 2020.

[12] A. S. Bedi, A. Koppel, and K. Rajawat, "Asynchronous online learning in multi-agent systems with proximity constraints," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 3, pp. 479–494, 2019.

[13] 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.

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

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

[16] 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.

[17] 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.

[18] 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.

[19] Z. Li, W. Shi, and M. Yan, "A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates," *IEEE Transactions on Signal Processing*, vol. 67, no. 17, pp. 4494–4506, 2019.

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

[21] 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.

[22] R. Xin and U. A. Khan, "A linear algorithm for optimization over directed graphs with geometric convergence," *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 315–320, 2018.

[23] Y. Sun, A. Daneshmand, and G. Scutari, "Convergence rate of distributed optimization algorithms based on gradient tracking," *arXiv preprint arXiv:1905.02637*, 2019.

[24] R. Xin, S. Kar, and U. A. Khan, "Gradient tracking and variance reduction for decentralized optimization and machine learning," *arXiv preprint arXiv:2002.05373*, 2020.

[25] D. Jakovetić, "A unification and generalization of exact distributed first-order methods," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 1, pp. 31–46, 2018.

[26] 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.

[27] D. Bajovic, D. Jakovetic, N. Krejic, and N. K. Jerinkic, "Newton-like method with diagonal correction for distributed optimization," *SIAM Journal on Optimization*, vol. 27, no. 2, pp. 1171–1203, 2017.

[28] F. Mansoori and E. Wei, "A fast distributed asynchronous Newton-based optimization algorithm," *IEEE Transactions on Automatic Control*, vol. 65, no. 7, pp. 2769–2784, 2020.

[29] N. K. Jerinkić, D. Jakovetić, N. Krejić, and D. Bajović, "Distributed second-order methods with increasing number of working nodes," *IEEE Transactions on Automatic Control*, vol. 65, no. 2, pp. 846–853, 2019.

[30] 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.

[31] ——, "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.

[32] 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.

[33] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton-Raphson consensus for distributed convex optimization," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 994–1009, 2015.

[34] S. Soori, K. Mishchenko, A. Mokhtari, M. M. Dehnavi, and M. Gurbuzbalaban, "DAve-QN: A distributed averaged quasi-newton method with local superlinear convergence rate," in *International Conference on Artificial Intelligence and Statistics*, 2020, pp. 1965–1976.

[35] J. Zhang, K. You, and T. Başar, "Distributed adaptive Newton methods with globally superlinear convergence," *arXiv preprint arXiv:2002.07378*, 2020.

[36] F. Yan, S. Sundaram, S. Vishwanathan, and Y. Qi, "Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2483–2493, 2012.

[37] A. Koppel, F. Y. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5149–5164, 2015.

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

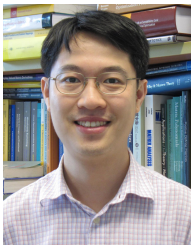
[39] S. U. Pillai, T. Suel, and S. Cha, "The Perron-Frobenius theorem: Some of its applications," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 62–75, 2005.

[40] M. Maros and J. Jaldén, "On the Q-linear convergence of distributed generalized ADMM under non-strongly convex function components," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 5, no. 3, pp. 442–453, 2019.

[41] V. Yadav and M. V. Salapaka, "Distributed protocol for determining when averaging consensus is reached," in *45th Annual Allerton Conf*, 2007, pp. 715–720.
[42] S. Giannini, A. Petitti, D. Di Paola, and A. Rizzo, "Asynchronous max-consensus protocol with time delays: Convergence results and applications," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 2, pp. 256–264, 2016.

Jiaojiao Zhang received the B.E. degree in automation from School of Automation, Harbin Engineering University, Harbin, China, in 2015. She received the master degree in control theory and control engineering from University of Science and Technology of China, Hefei, China, in 2018. She is currently pursuing the Ph.D. degree in the Department of Systems Engineering and Engineering Management, Chinese University of Hong Kong. She received the Hong Kong PhD Fellowship Scheme (HKPFS) in August 2018. Her current research interests include distributed optimization and algorithm design.

Qing Ling received the B.E. degree in automation and Ph.D. degree in control theory and control engineering from University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. He was a Postdoctoral Research Fellow with Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA, from 2006 to 2009 and an Associate Professor with Department of Automation, University of Science and Technology of China, from 2009 to 2017. He is currently a Professor with School of Computer Science and Engineering, Sun Yat-Sen University, Guangzhou, China. His current research interest includes distributed and decentralized optimization and its application in machine learning. He received the 2017 IEEE Signal Processing Society Young Author Best Paper Award as a supervisor. He is a Senior Area Editor of IEEE SIGNAL PROCESSING LETTERS.



Anthony Man-Cho So (M'12-SM'17) received the BSE degree in Computer Science from Princeton University, Princeton, NJ, USA, with minors in Applied and Computational Mathematics, Engineering and Management Systems, and German Language and Culture. He then received the M.Sc. degree in Computer Science and the Ph.D. degree in Computer Science with a Ph.D. minor in Mathematics from Stanford University, Stanford, CA, USA.

Dr. So joined The Chinese University of Hong Kong (CUHK) in 2007. He is now the Associate Dean of Student Affairs in the Faculty of Engineering, Deputy Master of Morningside College, and Professor in the Department of Systems Engineering and Engineering Management. His research focuses on optimization theory and its applications in various areas of science and engineering, including computational geometry, machine learning, signal processing, and statistics.

Dr. So is appointed as an Outstanding Fellow of the Faculty of Engineering at CUHK in 2019. He has received a number of research and teaching awards, including the 2018 IEEE Signal Processing Society Best Paper Award, the 2015 IEEE Signal Processing Society Signal Processing Magazine Best Paper Award, the 2014 IEEE Communications Society Asia-Pacific Outstanding Paper Award, the 2013 CUHK Vice-Chancellor's Exemplary Teaching Award, and the 2010 Institute for Operations Research and the Management Sciences (INFORMS) Optimization Society Optimization Prize for Young Researchers. He currently serves on the editorial boards of Journal of Global Optimization, Optimization Methods and Software, and SIAM Journal on Optimization. He was also the Lead Guest Editor of IEEE SIGNAL PROCESSING MAGAZINE Special Issue on "Non-Convex Optimization for Signal Processing and Machine Learning".