2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Accelerated Distributed Nesterov Gradient Descent for Convex and Smooth Functions

Guannan Qu, Na Li

***Abstract*— This paper considers the distributed optimization problem over a network, where the objective is to optimize a global function formed by an average of local functions, using only local computation and communication. We develop an Accelerated Distributed Nesterov Gradient Descent (Acc-DNGD) method for convex and smooth objective functions. We show that it achieves a $O(1/t^{1.4-\epsilon})$ ($\forall \epsilon \in (0, 1.4)$) convergence rate when a vanishing step size is used. The convergence rate can be improved to $O(1/t^2)$ when we use a fixed step size and the objective functions satisfy a special property. To the best of our knowledge, Acc-DNGD is the fastest among all distributed gradient-based algorithms that have been proposed so far.**

I. Introduction

Given a set of agents $\mathcal{N} = \{1, 2, \ldots, n\}$, each of which has a local convex cost function $f_i(x) : \mathbb{R}^N \to \mathbb{R}$, the objective of distributed optimization is to find x that minimizes the average of all the functions,

$$\min_{x \in \mathbb{R}^N} f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x)$$

using local communication and local computation. The local communication is defined through a connected and undirected communication graph $\mathcal{G} = (V, E)$, where the nodes $V = \mathcal{N}$ and edges $E \subset V \times V$. This problem has found various applications in multi-agent control, distributed state estimation over sensor networks, large scale computation in machine learning, etc [1]–[3].

There exist many studies on developing distributed algorithms for this problem, e.g., [4]–[15], most of which are distributed gradient descent algorithms. Each iteration is composed of a consensus step and a gradient descent step. These methods have achieved sublinear convergence rates (usually $O(\frac{1}{\sqrt{t}})$ for convex functions. When the functions are nonsmooth, the sublinear convergence rates match Centralized Gradient Descent (CGD). More recent work have improved these results for smooth functions, by adding a correction term [16]–[18], or using a gradient estimation sequence [19]–[25]. With these techniques, paper [16], [22] can achieve a $O(\frac{1}{t})$ convergence rate for smooth functions, matching the rate of CGD. Additionally, if strong convexity is further assumed, paper [16]–[18], [22]–[25] can achieve a linear convergence rate, matching the rate of CGD as well.

It is known that among all centralized gradient based algorithms, Centralized Nesterov Gradient Descent (CNGD) [26] achieves the optimal convergence rate in terms of first-order oracle complexity. For μ-strongly convex and L-smooth functions, it achieves a $O((1 - \sqrt{\mu/L})^t)$ convergence rate; for convex and L-smooth functions, it achieves a $O(1/t^2)$ convergence rate. The nice convergence rates lead to the question of this paper: how to decentralize the Nesterov Gradient method to achieve similar convergence rates? Our recent work [27] has studied the μ-strongly convex and L-smooth case. This paper will focus on the convex and L-smooth case (without the strongly convex assumption). Previous work in this line includes [28] that develops Distributed Nesterov Gradient (D-NG) method and shows that it has a convergence rate of $O(\frac{\log t}{t})$,[1] which is not faster than the rate of CGD ($O(\frac{1}{t})$).

In this paper, we propose an Accelerated Distributed Nesterov Gradient Descent (Acc-DNGD) method. We show that it achieves a $O(1/t^{1.4-\epsilon})$ (for any $\epsilon \in (0, 1.4)$) convergence rate when a vanishing step size is used. We further show that the convergence rate can be improved to $O(1/t^2)$ when we use a fixed step size and the objective function is a composition of a linear map and a strongly-convex and smooth function. Both rates are faster than what CGD and CGD-based distributed methods can achieve ($O(1/t)$). To the best of the authors' knowledge, the $O(1/t^{1.4-\epsilon})$ rate is the fastest among all distributed gradient-based algorithms being proposed so far.[2]

Our algorithm is a combination of CNGD and a gradient estimation scheme. The gradient estimation scheme has been studied under various contexts in [19]–[25]. As [22] has pointed out, when combining the gradient estimation scheme with a centralized algorithm, the resulting distributed algorithm could potentially match the convergence rate of the centralized algorithm. The results in this paper show that, although combining the scheme with CNGD will not give a convergence rate ($O(1/t^{1.4-\epsilon})$) matching that of CNGD ($O(1/t^2)$), it does improve over previously known CGD-based distributed algorithms ($O(1/t)$).

In the rest of the paper, Section II formally defines the problem and presents our algorithm and results. Section III proves the convergence rates. Section IV provides numerical simulations and Section V concludes the paper.

Notations. In this paper, n is the number of agents, and N is the dimension of the domain of the f_i's. Notation $\|\cdot\|$ denotes 2-norm for vectors, and Frobenius norm for matrices. Notation $\|\cdot\|_*$ denotes spectral norm for matrices. Notation $\langle \cdot, \cdot \rangle$ denotes inner product for vectors. Notation $\rho(\cdot)$ denotes spectral radius for square matrices, and $\mathbf{1}$ denotes a n-dimensional all one column vector. All vectors, when having dimension N (the dimension of the domain of the f_i's), will

Guannan Qu and Na Li are affiliated with John A. Paulson School of Engineering and Applied Sciences at Harvard University. Email: gqu@g.harvard.edu, nali@seas.harvard.edu. This work is supported under NSF ECCS 1608509 and NSF CAREER 1553407.

[1] Reference [28] also studies an algorithm that uses multiple consensus steps per iteration, and achieves a $O(1/t^2)$ convergence rate. In this paper, we focus on algorithms that only use one or a constant number of consensus steps per iteration.

[2] We only include algorithms that are gradient based (without extra information like Hessian), and use one (or a constant number of) step(s) of consensus after each gradient evaluation.

be regarded as row vectors. As a special case, all gradients, $\nabla f_i(x)$ and $\nabla f(x)$ are interpreted as N-dimensional row vectors. Notation "$\leq$", when applied to vectors of the same dimension, denotes element wise "less than or equal to".

II. Problem and Algorithm

A. Problem Formulation

Consider n agents, $\mathcal{N} = \{1, 2, \ldots, n\}$, each of which has a function $f_i : \mathbb{R}^N \to \mathbb{R}$. The objective of distributed optimization is to find x to minimize the average of all the functions, i.e.

$$\min_{x \in \mathbb{R}^N} f(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) \tag{1}$$

using local communication and local computation. The local communication is defined through a *connected undirected* communication graph $\mathcal{G} = (V, E)$, where the nodes $V = \mathcal{N}$ and the edges $E \subset V \times V$. Agent i and j can send information to each other if and only if $(i, j) \in E$. The local computation means that each agent can only make its decision based on the local function f_i and the information obtained from its neighbors.

Throughout the paper, we assume that f has a minimizer x^* with optimal value f^*. We will use the following assumptions in the rest of the paper.

Assumption 1. *$\forall i \in \mathcal{N}$, f_i is convex. As a result, f is also convex.*

Assumption 2. *$\forall i \in \mathcal{N}, f_i$ is L-smooth, that is, f_i is differentiable and the gradient is L-Lipschitz continuous, i.e., $\forall x, y \in \mathbb{R}^N$, $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$. As a result, f is L-smooth.*

Assumption 3. *The set of minimizers of f is compact.*

B. Centralized Nesterov Gradient Descent (CNGD)

We briefly introduce a version of centralized Nesterov Gradient Descent (CNGD) that is derived from Section 2.2 of [26]. CNGD keeps updating three variables $x(t), v(t), y(t) \in \mathbb{R}^N$, starting from an initial point $x(0) = v(0) = y(0) \in \mathbb{R}^N$, and the update equation is given by

$$x(t+1) = y(t) - \eta \nabla f(y(t)) \tag{2a}$$

$$v(t+1) = v(t) - \frac{\eta}{\alpha_t} \nabla f(y(t)) \tag{2b}$$

$$y(t+1) = (1 - \alpha_{t+1}) x(t+1) + \alpha_{t+1} v(t+1), \tag{2c}$$

where $(\alpha_t)_{t=0}^{\infty}$ is defined by an arbitrarily chosen $\alpha_0 \in (0, 1)$ and the update equation $\alpha_{t+1}^2 = (1 - \alpha_{t+1})\alpha_t^2$, where α_{t+1} always takes the unique solution in $(0, 1)$. The following theorem (adapted from [26, Thm 2.2.1, Lem. 2.2.4]) gives the convergence rate of CNGD.

Theorem 1. *In CNGD (2), under Assumption 1 and 2, when $0 < \eta \leq \frac{1}{L}$, we have $f(x(t)) - f^* = O(\frac{1}{t^2})$.*

C. Our Algorithm: Accelerated Distributed Nesterov Gradient Descent (Acc-DNGD)

We design our algorithm based on a consensus matrix $W = [w_{ij}] \in \mathbb{R}^{n \times n}$. Here w_{ij} stands for how much agent i weighs its neighbor j's information. W satisfies the following properties:

(a) $\forall (i, j) \in E$, $w_{ij} > 0$. $\forall i$, $w_{ii} > 0$. $w_{ij} = 0$ elsewhere.

(b) Matrix W is doubly stochastic, i.e. $\sum_{i'} w_{i'j} = \sum_{j'} w_{ij'} = 1$ for all $i, j \in \mathcal{N}$.

As a result, $\exists \sigma \in (0, 1)$ which depends on the spectrum of W, such that for any $\omega \in \mathbb{R}^{n \times 1}$, we have the "averaging property", $\|W\omega - \mathbf{1}\bar{\omega}\| \leq \sigma \|\omega - \mathbf{1}\bar{\omega}\|$ where $\bar{\omega} = \frac{1}{n}\mathbf{1}^T\omega$ (the average of the entries in ω) [29]. How to select a consensus matrix to satisfy these properties has been intensely studied, e.g. [29], [30].

In our algorithm Acc-DNGD, each agent keeps a copy of the three variables in CNGD, $x_i(t)$, $v_i(t)$, $y_i(t)$ and in addition $s_i(t)$ which serves as a gradient estimator. The initial condition is $x_i(0) = v_i(0) = y_i(0) = 0$ and $s_i(0) = \nabla f(0)$,[3] and the algorithm updates as follows:

$$x_i(t+1) = \sum_j w_{ij} y_j(t) - \eta_t s_i(t) \tag{3a}$$

$$v_i(t+1) = \sum_j w_{ij} v_j(t) - \frac{\eta_t}{\alpha_t} s_i(t) \tag{3b}$$

$$y_i(t+1) = (1 - \alpha_{t+1}) x_i(t+1) + \alpha_{t+1} v_i(t+1) \tag{3c}$$

$$s_i(t+1) = \sum_j w_{ij} s_j(t) + \nabla f_i(y_i(t+1)) - \nabla f_i(y_i(t)) \tag{3d}$$

where $[w_{ij}]_{n \times n}$ are the consensus weights and $\eta_t \in (0, \frac{1}{L})$ are the step sizes. Sequence $(\alpha_t)_{t \geq 0}$ is generated by, first selecting $\alpha_0 = \sqrt{\eta_0 L} \in (0, 1)$, then given $\alpha_t \in (0, 1)$, selecting α_{t+1} to be the unique solution in $(0, 1)$ of the following equation,[4]

$$\alpha_{t+1}^2 = \frac{\eta_{t+1}}{\eta_t}(1 - \alpha_{t+1})\alpha_t^2.$$

We will consider two variants of the algorithm with the following two step size rules.

- **Vanishing step size rule:** $\eta_t = \eta \frac{1}{(t+t_0)^\beta}$ for some $\eta \in (0, \frac{1}{L})$, $\beta \in (0, 2)$ and $t_0 \geq 1$.
- **Fixed step size rule:** $\eta_t = \eta > 0$.

Because $w_{ij} = 0$ when $(i, j) \notin E$, each node i only needs to send $x_i(t)$, $v_i(t)$, $y_i(t)$ and $s_i(t)$ to its neighbors. Therefore, the algorithm can be operated in a fully distributed fashion with only local communication. The additional term $s_i(t)$ allows each agent to obtain an estimate on the global gradient $\frac{1}{n}\sum_j \nabla f_j(y_j(t))$ (for more details, see Section II-D). Compared with distributed algorithms without this estimation term, it helps improve the convergence speed. As a result, we call this method as Accelerated Distributed Nesterov Gradient Descent (Acc-DNGD) method.

D. Intuition Behind our Algorithm

Here we briefly explain how the algorithm works. First we note that Eq. (3a)-(3c) is similar to Eq. (2), except the "weighted average terms" ($\sum_j w_{ij} y_j(t)$, $\sum_j w_{ij} v_j(t)$) and the new term $s_i(t)$ that replaces the gradient terms.

[3] We note that the initial condition $s_i(0) = \nabla f(0)$ requires the agents to conduct an initial run of consensus. We impose this initial condition for technical reasons, while we expect the results of this paper to hold for a relaxed initial condition, $s_i(0) = \nabla f_i(0)$ which does not need initial coordination. We use the relaxed initial condition in numerical simulations.

[4] Without causing any confusion with the α_t in (2), in the rest of the paper we abuse the notation of α_t.

We have the following circular arguments that explain why algorithm (3) should work.

Argument 1: Assuming $s_i(t) \approx \frac{1}{n}\sum_{j=1}^n \nabla f_j(y_j(t))$, then the algorithm converges.

To see this, notice the "weighted average terms" $(\sum_j w_{ij}y_j(t),\ \sum_j w_{ij}v_j(t))$ ensure that different agents reach "consensus", i.e. $\forall j,\ x_i(t) \approx x_j(t), y_i(t) \approx y_j(t)$ and $v_i(t) \approx v_j(t)$ (as a result, $\sum_j w_{ij}y_j(t) \approx y_i(t)$, $\sum_j w_{ij}v_j(t) \approx v_i(t)$). If we further assume that $s_i(t) \approx \frac{1}{n}\sum_j \nabla f_j(y_j(t))$, then since $y_j(t) \approx y_i(t)$, we have $s_i(t) \approx \frac{1}{n}\sum_j \nabla f_j(y_i(t)) = \nabla f(y_i(t))$. Hence (3a)-(3c) can be rewritten as

$$x_i(t+1) \approx y_i(t) - \eta_t \nabla f(y_i(t)) \quad (4a)$$

$$v_i(t+1) \approx v_i(t) - \frac{\eta_t}{\alpha_t}\nabla f(y_i(t)) \quad (4b)$$

$$y_i(t+1) \approx (1-\alpha_{t+1})x_i(t+1) + \alpha_{t+1}v_i(t+1) \quad (4c)$$

which is exactly (2) (except the step size rule), and hence we expect convergence. □

Argument 2: Assuming the algorithm converges, then $s_i(t) \approx \frac{1}{n}\sum_{j=1}^n \nabla f_j(y_j(t))$.

To see this, from (3d) and the fact that $\sum_{i'} w_{i'j} = 1$, we have $\bar{s}(t) := \frac{1}{n}\sum_{j=1}^n s_j(t) = \frac{1}{n}\sum_{j=1}^n \nabla f_j(y_j(t))$. Assuming the convergence of the algorithm, we will have the input to (3d), $\|\nabla f_i(y_i(t+1)) - \nabla f_i(y_i(t))\| \le L\|y_i(t+1) - y_i(t)\| \to 0$. Because of the vanishing input, and the "taking weighted average of neighbor" step $(\sum_j w_{ij}s_j(t))$ in (3d), we can expact that eventually $s_i(t) \approx \bar{s}(t) = \frac{1}{n}\sum_{j=1}^n \nabla f_j(y_j(t))$. □

Though Argument 1 and 2 only form a circular argument, they provide a high-level guideline for the rigorous proof in Section III. To give the rigorous proof, it turns out that we need to use a vanishing step size in (3) instead of a fixed step size as in (2) (we can still use a fixed step size if f_i has special structures, cf. Theorem 3). This slows down the convergence rate of our algorithm $(1/t^{1.4-\epsilon})$ compared to CNGD $(1/t^2)$ (cf. Theorem 2).

An observation from the above circular argument is that, $s_i(t)$ acts as a "gradient estimator" that estimates the average gradient $\frac{1}{n}\sum_j \nabla f_j(y_j(t))$. This observation can be used to devise stopping criterion of the algorithm (e.g. the algorithm stops when $s_i(t)$ is sufficiently close to 0).

E. Convergence of the Algorithm

To state the convergence results, we need to define the average sequence, $\bar{x}(t) = \frac{1}{n}\sum_{i=1}^n x_i(t) \in \mathbb{R}^{1\times N}$. We summarize our convergence results below.

Theorem 2. *Suppose Assumption 1, 2 and 3 are true and without loss of generality we assume $\bar{v}(0) \neq x^*$. Let the step size be $\eta_t = \frac{\eta}{(t+t_0)^\beta}$ with $\beta = 0.6+\epsilon$ where $\epsilon \in (0, 1.4)$. Suppose the following conditions are met.*

(i)
$$t_0 > \frac{1}{\min((\frac{\sigma+3}{\sigma+2}\frac{3}{4})^{\sigma/(28\beta)}, (\frac{16}{15+\sigma})^{\frac{1}{\beta}}) - 1}.$$

(ii)
$$\eta < \min(\frac{\sigma^2}{9^3 L}, \frac{(1-\sigma)^4}{36866L}).$$

(iii)
$$\eta < \left(\frac{D(\beta,t_0)(\beta-0.6)(1-\sigma)^2}{9216(t_0+1)^{2-\beta}L^{2/3}[4+R^2/\|\bar{v}(0)-x^*\|^2]}\right)^{3/2}$$

where $D(\beta,t_0) = \frac{1}{(t_0+3)^2 e^{16+\frac{6}{2-\beta}}}$ and R is the diameter of the $(2f(\bar{x}(0)) - f^* + 2L\|\bar{v}(0)-x^*\|^2)$-level set of f.[5]

Then, $f(\bar{x}(t)) - f^* = O(\frac{1}{t^{1.4-\epsilon}})$.

In Theorem 2, condition (i) intends to make η_t/η_{t+1} close to 1 which is required in Lemma 6 (iii), and conditions (ii)(iii) intend to make η_t close to 0 which is required in Lemma 6 (ii). While the conditions are needed for the proof, we expect the same result will hold if we simply let $t_0 = 1$ and $\eta = \frac{1}{2L}$, which is what we choose in the simulations in Section IV. The reason is that, regardless of the value of η and t_0, we have $\eta_t \to 0$ and $\eta_t/\eta_{t+1} \to 1$, and hence for large enough t, η_t and η_t/η_{t+1} will automatically be close to 0 and 1 respectively.

While in Theorem 2 we require $\beta > 0.6$, we conjecture that the algorithm will still converge even if $\beta \in [0, 0.6]$ and the convergence rate will be $O(\frac{1}{t^{2-\beta}})$. We note that $\beta = 0$ corresponds to the case of a fixed step size. In Section IV we will use numerical methods to test this conjecture.

In the next theorem, we provide a $O(\frac{1}{t^2})$ convergence result when a fixed step size is used and the objective functions belong to a special class.

Theorem 3. *Assume each $f_i(x)$ can be written as $f_i(x) = h_i(xA_i)$, where A_i is a non-zero $N \times M_i$ matrix, and $h_i(x) : \mathbb{R}^{1\times M_i} \to \mathbb{R}$ is a μ_0-strongly convex and L_0-smooth function. Suppose we use the fixed step size rule $\eta_t = \eta$, with*

$$0 < \eta < \min(\frac{\sigma^2}{9^3 L}, \frac{\mu^{1.5}(1-\sigma)^3}{L^{2.5}3456^{1.5}})$$

where $L = L_0\nu$ with $\nu = \max_i \|A_i\|_*^2$ (where $\|A_i\|_*$ means the spectral norm of A_i); and $\mu = \mu_0\gamma$ with γ being the smallest non-zero eigenvalue of matrix $A = \frac{1}{n}\sum_{i=1}^n A_i A_i^T$. Then, we have $f(\bar{x}(t)) - f^* = O(\frac{1}{t^2})$.

An important example of the type of function $f_i(x)$ in Theorem 3 is the square error for linear regression when the sample size is less than the parameter dimension.

Remark 1. *All the step size conditions used in this section are conservative. This is because we have used coarse spectral bounds in the proofs (see Lemma 10, 11, 12), in order to simplify mathematical calculations. In numerical simulations, we show that large step sizes can be used. When applying the algorithm in practice, this may require trial and error to pre-tune the step size.*

III. Convergence Analysis

In this section, we will provide the proof of the convergence results. We will first provide a proof overview in Section III-A and then defer the detailed proof to the rest of the section. Due to space limit, we omit some proofs, which can be found in the full version of this paper [32].

A. Proof Overview

We introduce matrix notations $x(t)$, $v(t)$, $y(t)$, $s(t)$, $\nabla(t) \in \mathbb{R}^{n\times N}$ to simplify the mathematical expressions,[6]

$$x(t) = [x_1(t)^T, x_2(t)^T, \ldots, x_n(t)^T]^T$$

[5]Here we have used the fact that by Assumption 1 and 3, all level sets of f are bounded. See Proposition B.9 of [31].

[6]Without causing any confusion with notations in (2), in this section we abuse the use of notation $x(t), v(t), y(t)$.

$$v(t) = [v_1(t)^T, v_2(t)^T, \dots, v_n(t)^T]^T$$
$$y(t) = [y_1(t)^T, y_2(t)^T, \dots, y_n(t)^T]^T$$
$$s(t) = [s_1(t)^T, s_2(t)^T, \dots, s_n(t)^T]^T$$
$$\nabla(t) = [\nabla f_1(y_1(t))^T, \nabla f_2(y_2(t))^T, \dots, \nabla f_n(y_n(t))^T]^T.$$

Now our algorithm in (3) can be written as

$$x(t+1) = Wy(t) - \eta_t s(t) \quad (5a)$$
$$v(t+1) = Wv(t) - \frac{\eta_t}{\alpha_t} s(t) \quad (5b)$$
$$y(t+1) = (1-\alpha_{t+1})x(t+1) + \alpha_{t+1} v(t+1) \quad (5c)$$
$$s(t+1) = Ws(t) + \nabla(t+1) - \nabla(t). \quad (5d)$$

Apart from the average sequence $\bar{x}(t) = \frac{1}{n}\sum_{i=1}^n x_i(t) \in \mathbb{R}^{1\times N}$ that we have defined, we also define several other average sequences, $\bar{v}(t) = \frac{1}{n}\sum_{i=1}^n v_i(t)$, $\bar{y}(t) = \frac{1}{n}\sum_{i=1}^n y_i(t)$, $\bar{s}(t) = \frac{1}{n}\sum_{i=1}^n s_i(t)$, and $g(t) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(y_i(t))$.

Overview of the Proof. We derive a series of lemmas (Lemma 4, 5, 6 and 7) that will work for both the vanishing and the fixed step size case. We firstly derive the update formula for the average sequences (Lemma 4). Then, we show that the update rule for the average sequences is in fact centralized Nesterov Gradient Descent (CNGD) with inexact gradients [33], and the inexactness is characterized by "consensus error" $\|y(t) - \mathbf{1}\bar{y}(t)\|$ (Lemma 5). The consensus error is bounded in Lemma 6. Then, we apply the proof of CNGD (see e.g. [26]) to the average sequences in spite of the consensus error, and derive an intermediate result in Lemma 7. Lastly, we finish the proof of Theorem 2 in Section III-C. The proof of Theorem 3 can be found in Appendix-F in [32].

Lemma 4. *The following equalities hold.*

$$\bar{x}(t+1) = \bar{y}(t) - \eta_t g(t) \quad (6a)$$
$$\bar{v}(t+1) = \bar{v}(t) - \frac{\eta_t}{\alpha_t} g(t) \quad (6b)$$
$$\bar{y}(t+1) = (1-\alpha_{t+1})\bar{x}(t+1) + \alpha_{t+1}\bar{v}(t+1) \quad (6c)$$
$$\bar{s}(t+1) = \bar{s}(t) + g(t+1) - g(t) = g(t+1) \quad (6d)$$

Proof: We omit the proof since these equalities can be easily derived using the fact that W is doubly stochastic. For (6d) we also need to use the fact that $\bar{s}(0) = g(0)$. □

From (6a)-(6c) we see that the sequences $\bar{x}(t)$, $\bar{v}(t)$ and $\bar{y}(t)$ follow a update rule similar to the CNGD in (2). The only difference is that the $g(t)$ in (6a)-(6c) is not the exact gradient $\nabla f(\bar{y}(t))$ in CNGD. In the following Lemma, we show that $g(t)$ is an inexact gradient.[7]

Lemma 5. *Under Assumption 1, 2, $\forall t$, $g(t)$ is an inexact gradient of f at $\bar{y}(t)$ with error $O(\|y(t) - \mathbf{1}\bar{y}(t)\|^2)$ in the sense that, $\forall \omega \in \mathbb{R}^N$,*

$$f(\omega) \geq \hat{f}(t) + \langle g(t), \omega - \bar{y}(t)\rangle \quad (7)$$
$$f(\omega) \leq \hat{f}(t) + \langle g(t), \omega - \bar{y}(t)\rangle + L\|\omega - \bar{y}(t)\|^2 + L\frac{1}{n}\|y(t) - \mathbf{1}\bar{y}(t)\|^2, \quad (8)$$

where $\hat{f}(t) = \frac{1}{n}\sum_{i=1}^n [f_i(y_i(t)) + \langle \nabla f_i(y_i(t)), \bar{y}(t) - y_i(t)\rangle]$.

[7]For more information regarding why (7) (8) define an "inexact gradient", we refer the readers to [33].

Proof: We omit the proof and refer the readers to [27, Lem. 4]. □

The consensus error $\|y(t) - \mathbf{1}\bar{y}(t)\|$ in the previous lemma is bounded by the following lemma whose proof is given in Section III-B.

Lemma 6. *Suppose the step sizes satisfy*

(i) $\eta_t \geq \eta_{t+1} > 0$,
(ii) $\eta_0 < \min(\frac{\sigma^2}{9^3 L}, \frac{(1-\sigma)^3}{6144L})$,
(iii) $\sup_{t\geq 0} \frac{\eta_t}{\eta_{t+1}} \leq \min((\frac{\sigma+3}{\sigma+2}\frac{3}{4})^{\sigma/28}, \frac{16}{15+\sigma})$.

Then, under Assumption 2, we have,

$$\|y(t) - \mathbf{1}\bar{y}(t)\| \leq \kappa\sqrt{n}\chi_2(\eta_t)\left[L\|\bar{y}(t) - \bar{x}(t)\| + \frac{8}{1-\sigma}L\eta_t\|g(t)\|\right]$$

where $\chi_2 : \mathbb{R} \to \mathbb{R}$ is a function satisfying $0 < \chi_2(\eta_t) \leq \frac{2}{L^{2/3}}\eta_t^{1/3}$, and $\kappa = \frac{6}{(1-\sigma)}$.

We next provide the following intermediate result. The proof roughly follows the same steps of [26, Lemma 2.2.3], and can be found in Appendix-D of [32].

Lemma 7. *Define $\gamma_0 = \frac{\alpha_0^2}{\eta_0(1-\alpha_0)} = \frac{L}{1-\alpha_0}$. We define a series of functions $(\Phi_t : \mathbb{R}^N \to \mathbb{R})_{t\geq 0}$, with*

$$\Phi_0(\omega) = f(\bar{x}(0)) + \frac{\gamma_0}{2}\|\omega - \bar{v}(0)\|^2$$

and

$$\Phi_{t+1}(\omega) = (1-\alpha_t)\Phi_t(\omega) + \alpha_t[\hat{f}(t) + \langle g(t), \omega - \bar{y}(t)\rangle]. \quad (9)$$

Then, under Assumption 1 and 2, the following holds.

(i) *We have,*

$$\Phi_t(\omega) \leq f(\omega) + \lambda_t(\Phi_0(\omega) - f(\omega)) \quad (10)$$

where λ_t is defined through $\lambda_0 = 1$, and $\lambda_{t+1} = (1-\alpha_t)\lambda_t$.

(ii) Function $\Phi_t(\omega)$ can be written as

$$\Phi_t(\omega) = \phi_t^* + \frac{\gamma_t}{2}\|\omega - \bar{v}(t)\|^2 \quad (11)$$

where γ_t is defined through $\gamma_{t+1} = \gamma_t(1-\alpha_t)$, and ϕ_t^* is some real number that satisfies $\phi_0^* = f(\bar{x}(0))$, and

$$\phi_{t+1}^* = (1-\alpha_t)\phi_t^* + \alpha_t\hat{f}(t) - \frac{1}{2}\eta_t\|g(t)\|^2 + \alpha_t\langle g(t), \bar{v}(t) - \bar{y}(t)\rangle. \quad (12)$$

B. Proof of the Bounded Consensus Error (Lemma 6)

We will frequently use the following lemmas, whose proofs can be found in Appendix-A of [32].

Lemma 8. *The following equalities are true.*

$$\bar{y}(t+1) - \bar{y}(t) = \alpha_{t+1}(\bar{v}(t) - \bar{y}(t)) - \eta_t\left[\frac{\alpha_{t+1}}{\alpha_t} + 1 - \alpha_{t+1}\right]g(t) \quad (13)$$

$$\bar{v}(t+1) - \bar{y}(t+1) = (1-\alpha_{t+1})(\bar{v}(t) - \bar{y}(t)) + \eta_t(1-\alpha_{t+1})(1 - \frac{1}{\alpha_t})g(t) \quad (14)$$

Lemma 9. *Under Assumption 2, the following are true.*

$$\|\nabla(t+1) - \nabla(t)\| \leq L\|y(t+1) - y(t)\| \quad (15)$$
$$\|g(t) - \nabla f(\bar{y}(t))\| \leq \frac{L}{\sqrt{n}}\|y(t) - \mathbf{1}\bar{y}(t)\| \quad (16)$$

Proof of Lemma 6:

Overview of the proof. The proof is divided into three steps. In step 1, we treat the algorithm (5) as a linear system and derive a linear system inequality (17). In step 2, we analyze the state transition matrix in (17) and prove a few spectral properties. In step 3, we further analyze the linear system (17) and bound the state by the input, from which the conclusion of the lemma follows. Throughout the proof, we will frequently use an easy-to-check fact: α_t is a decreasing sequence.

Step 1: A Linear System Inequality. Define $z(t) = [\alpha_t\|v(t) - \mathbf{1}\bar{v}(t)\|, \|y(t) - \mathbf{1}\bar{y}(t)\|, \|s(t) - \mathbf{1}g(t)\|]^T \in \mathbb{R}^3$, $b(t) = [0, 0, \sqrt{n}a(t)]^T \in \mathbb{R}^3$ where

$$a(t) \triangleq \alpha_t L\|\bar{v}(t) - \bar{y}(t)\| + 2\lambda L\eta_t\|g(t)\|$$

in which $\lambda \triangleq \frac{4}{1-\sigma} > 1$. The desired inequality is (17). The proof of (17) is similar to that of [27, Eq. (8)]. Due to space limit, it is omitted and can be found in Appendix-B of [32].

$$z(t+1) \le \overbrace{\begin{bmatrix} \sigma & 0 & \eta_t \\ \sigma & \sigma & 2\eta_t \\ L & 2L & \sigma + 2\eta_t L \end{bmatrix}}^{\triangleq G(\eta_t)} z(t) + b(t) \qquad (17)$$

Step 2: Spectral Properties of $G(\cdot)$. When η is positive, $G(\eta)$ is a nonnegative matrix and $G(\eta)^2$ is a positive matrix. By Perron-Frobenius Theorem [34, Thm. 8.5.1] $G(\eta)$ has a unique largest (in magnitude) eigenvalue that is a positive real with multiplicity 1, and the eigenvalue is associated with an eigenvector with positive entries. We let the unique largest eigenvalue be $\theta(\eta) = \rho(G(\eta))$ and let its eigenvector be $\chi(\eta) = [\chi_1(\eta), \chi_2(\eta), \chi_3(\eta)]^T$, normalized by $\chi_3(\eta) = 1$. We give bounds on the eigenvalue and the eigenvector in the following lemmas, whose proofs can be found in Appendix-C of [32].

Lemma 10. *When $0 < \eta L < 1$, we have $\sigma < \theta(\eta) < \sigma + 4(\eta L)^{1/3}$, and $\chi_2(\eta) \le \frac{2}{L^{2/3}}\eta^{1/3}$.*

Lemma 11. *When $\eta \in (0, \frac{\sqrt{\sigma}}{L2\sqrt{2}})$, $\theta(\eta) \ge \sigma + (\sigma\eta L)^{1/3}$ and $\chi_1(\eta) < \frac{\eta}{(\sigma\eta L)^{1/3}}$.*

Lemma 12. *When $\zeta_1, \zeta_2 \in (0, \frac{\sigma^2}{9^3 L})$, then $\frac{\chi_1(\zeta_1)}{\chi_1(\zeta_2)} \le \max((\frac{\zeta_2}{\zeta_1})^{6/\sigma}, (\frac{\zeta_1}{\zeta_2})^{6/\sigma})$ and $\frac{\chi_2(\zeta_1)}{\chi_2(\zeta_2)} \le \max((\frac{\zeta_2}{\zeta_1})^{28/\sigma}, (\frac{\zeta_1}{\zeta_2})^{28/\sigma})$.*

It is easy to check that, under our step size condition (ii), all the conditions of Lemma 10, 11, 12 are satisfied.

Step 3: Bound the state by the input. With the above preparations, now we prove, by induction, the following statement,

$$z(t) \le \sqrt{n}a(t)\kappa\chi(\eta_t) \qquad (18)$$

where $\kappa = \frac{6}{1-\sigma}$. Equation (18) is true for $t = 0$, since the left hand side is zero when $t = 0$. Assume (18) holds for t. We now show (18) is true for $t + 1$. We divide the rest of the proof into two sub-steps. Briefly speaking, step 3.1 proves that the input to the system (17), $a(t + 1)$ does not decrease too much compared to $a(t)$ ($a(t + 1) \ge \frac{\sigma+3}{4}a(t)$); while step 3.2 shows that the state $z(t + 1)$, compared to $z(t)$, decreases enough for (18) to hold for $t + 1$.

Step 3.1: We prove that $a(t + 1) \ge \frac{\sigma+3}{4}a(t)$. By (14),

$$\begin{aligned}
&a(t+1)\\
&= \alpha_{t+1}L\|(1-\alpha_{t+1})(\bar{v}(t) - \bar{y}(t))\\
&\qquad + (1-\alpha_{t+1})(1 - \frac{1}{\alpha_t})\eta_t g(t)\| + 2\lambda\eta_{t+1}L\|g(t+1)\|\\
&\ge \alpha_{t+1}(1-\alpha_{t+1})L\|\bar{v}(t) - \bar{y}(t)\|\\
&\qquad - \frac{\alpha_{t+1}}{\alpha_t}(1-\alpha_{t+1})(1-\alpha_t)\eta_t L\|g(t)\|\\
&\qquad + 2\lambda\eta_{t+1}L\|g(t)\| - 2\lambda\eta_{t+1}L\|g(t+1) - g(t)\|.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&a(t) - a(t+1)\\
&\le \Big[\alpha_t - \alpha_{t+1}(1-\alpha_{t+1})\Big]L\|\bar{v}(t) - \bar{y}(t)\|\\
&\quad + \Big[\frac{\alpha_{t+1}}{\alpha_t}(1-\alpha_{t+1})(1-\alpha_t)\eta_t L\\
&\quad + 2\lambda\eta_t L - 2\lambda\eta_{t+1}L\Big]\|g(t)\| + 2\lambda\eta_{t+1}L\|g(t+1) - g(t)\|\\
&\le \Big[\alpha_t - \alpha_{t+1}(1-\alpha_{t+1})\Big]L\|\bar{v}(t) - \bar{y}(t)\|\\
&\qquad + (\eta_t + 2\lambda(\eta_t - \eta_{t+1}))L\|g(t)\|\\
&\qquad + 2\lambda\eta_{t+1}L\|g(t+1) - g(t)\|\\
&\le \max(1 - \frac{\alpha_{t+1}}{\alpha_t} + \frac{\alpha_{t+1}^2}{\alpha_t}, \frac{1}{2\lambda} + \frac{\eta_t - \eta_{t+1}}{\eta_t})a(t)\\
&\qquad + 2\lambda\eta_{t+1}L\|g(t+1) - g(t)\| \qquad (19)
\end{aligned}$$

where in the last inequality, we have used the elementary fact that for four positive numbers a_1, a_2, a_3, a_4 and $x, y \ge 0$, we have $a_1x + a_2y = \frac{a_1}{a_3}a_3x + \frac{a_2}{a_4}a_4y \le \max(\frac{a_1}{a_3}, \frac{a_2}{a_4})(a_3x + a_4y)$

Next, we expand $\|g(t + 1) - g(t)\|$,

$$\begin{aligned}
&\|g(t+1) - g(t)\|\\
&\le \|g(t+1) - \nabla f(\bar{y}(t+1))\| + \|g(t) - \nabla f(\bar{y}(t))\|\\
&\qquad + \|\nabla f(\bar{y}(t+1)) - \nabla f(\bar{y}(t))\|\\
&\overset{(a)}{\le} \frac{L}{\sqrt{n}}\|y(t+1) - \mathbf{1}\bar{y}(t+1)\| + \frac{L}{\sqrt{n}}\|y(t) - \mathbf{1}\bar{y}(t)\|\\
&\qquad + L\|\bar{y}(t+1) - \bar{y}(t)\|\\
&\overset{(b)}{\le} \frac{L}{\sqrt{n}}\sigma\alpha_t\|v(t) - \mathbf{1}\bar{v}(t)\| + \frac{L}{\sqrt{n}}2\|y(t) - \mathbf{1}\bar{y}(t)\|\\
&\qquad + \frac{L}{\sqrt{n}}2\eta_t\|s(t) - \mathbf{1}g(t)\| + a(t)\\
&\overset{(c)}{\le} L\sigma\kappa\chi_1(\eta_t)a(t) + 2L\kappa\chi_2(\eta_t)a(t)\\
&\qquad + 2L\eta_t\kappa\chi_3(\eta_t)a(t) + a(t)\\
&\overset{(d)}{\le} a(t)\Big\{L\sigma\kappa\frac{\eta_t}{(\sigma\eta_t L)^{1/3}} + 2L\kappa\frac{2\eta_t^{1/3}}{L^{2/3}} + 2L\eta_t\kappa + 1\Big\}\\
&\overset{(e)}{\le} 8\kappa a(t). \qquad (20)
\end{aligned}$$

Here (a) is due to (16); (b) is due to the second row of (17) and the fact that $a(t) \ge L\|\bar{y}(t+1) - \bar{y}(t)\|$; (c) is due to the induction assumption (18). In (d), we have used the bound on $\chi_1(\cdot)$ (Lemma 11), $\chi_2(\cdot)$ (Lemma 10), and $\chi_3(\eta_t) = 1$. In (e), we have used $\eta_t L < 1$, $\sigma < 1$ and $\kappa > 1$.

Combining (20) with (19), we have

$$\begin{aligned}
&a(t) - a(t+1)\\
&\le \max(1 - \frac{\alpha_{t+1}}{\alpha_t} + \frac{\alpha_{t+1}^2}{\alpha_t}, \frac{1}{2\lambda} + \frac{\eta_t - \eta_{t+1}}{\eta_t})a(t)\\
&\qquad + 16\kappa\lambda\eta_{t+1}La(t)\\
&\le \Big[\max(1 - \frac{\eta_{t+1}}{\eta_t} + 2\alpha_{t+1}, \frac{1-\sigma}{8} + \frac{\eta_t - \eta_{t+1}}{\eta_t})
\end{aligned}$$

$$+\frac{384}{(1-\sigma)^2}\eta_0 L\Big]a(t)$$

where in the last inequality, we have used the fact that

$$1-\frac{\alpha_{t+1}}{\alpha_t}+\frac{\alpha_{t+1}^2}{\alpha_t}<1-\frac{\alpha_{t+1}^2}{\alpha_t^2}+\alpha_{t+1}$$
$$=1-\frac{\eta_{t+1}}{\eta_t}(1-\alpha_{t+1})+\alpha_{t+1}<1-\frac{\eta_{t+1}}{\eta_t}+2\alpha_{t+1}.$$

By the step size condition (iii), $\frac{\eta_t}{\eta_{t+1}}\leq\frac{16}{15+\sigma}$, and hence $1-\frac{\eta_{t+1}}{\eta_t}\leq\frac{1-\sigma}{16}$. By the step size condition (ii), $2\alpha_{t+1}\leq 2\alpha_0=2\sqrt{\eta_0 L}\leq\frac{1-\sigma}{16}$, and $\eta_0 L\frac{384}{(1-\sigma)^2}<\frac{1-\sigma}{16}$. Combining the above, we have $a(t)-a(t+1)\leq\frac{1-\sigma}{4}a(t)$. Hence $a(t+1)\geq\frac{3+\sigma}{4}a(t)$.

Step 3.2: Finishing the induction. We have,

$$\begin{aligned}
z(t+1)&\overset{(a)}{\leq} G(\eta_t)z(t)+b(t)\\
&\overset{(b)}{\leq} G(\eta_t)\sqrt{n}a(t)\kappa\chi(\eta_t)+\sqrt{n}a(t)\chi(\eta_t)\\
&\overset{(c)}{=}\theta(\eta_t)\sqrt{n}a(t)\kappa\chi(\eta_t)+\sqrt{n}a(t)\chi(\eta_t)\\
&=\sqrt{n}a(t)\chi(\eta_t)(\kappa\theta(\eta_t)+1)\\
&\overset{(d)}{\leq}\sqrt{n}a(t+1)\chi(\eta_{t+1})(\kappa\frac{\sigma+1}{2}+1)\frac{4}{3+\sigma}\\
&\quad\times\max(\frac{\chi_1(\eta_t)}{\chi_1(\eta_{t+1})},\frac{\chi_2(\eta_t)}{\chi_2(\eta_{t+1})},1)\\
&\overset{(e)}{=}\sqrt{n}a(t+1)\chi(\eta_{t+1})\frac{\sigma+2}{3}\kappa\frac{4}{\sigma+3}\\
&\quad\times\max(\frac{\chi_1(\eta_t)}{\chi_1(\eta_{t+1})},\frac{\chi_2(\eta_t)}{\chi_2(\eta_{t+1})},1)\\
&\overset{(f)}{\leq}\sqrt{n}a(t+1)\kappa\chi(\eta_{t+1})
\end{aligned}\tag{21}$$

where (a) is due to (17), and (b) is due to induction assumption (18), and (c) is because $\theta(\eta_t)$ is an eigenvalue of $G(\eta_t)$ with eigenvector $\chi(\eta_t)$, and (d) is due to step 3.1, and $\theta(\eta_t)<\sigma+4(\eta_0 L)^{1/3}<\frac{1+\sigma}{2}$ (by step size condition (ii) and Lemma 10), and in (e), we have used by the definition of κ, $\kappa\frac{\sigma+1}{2}+1=\frac{\sigma+2}{3}\kappa$. For (f), we have used that by Lemma 12 and step size condition (iii),

$$\max(\frac{\chi_1(\eta_t)}{\chi_1(\eta_{t+1})},\frac{\chi_2(\eta_t)}{\chi_2(\eta_{t+1})},1)\leq(\frac{\eta_t}{\eta_{t+1}})^{28/\sigma}\leq\frac{\sigma+3}{\sigma+2}\frac{3}{4}.$$

Now, (18) is proven for $t+1$, and hence is true for all t. Therefore, we have

$$\|y(t)-\mathbf{1}\bar{y}(t)\|\leq\kappa\sqrt{n}a(t)\chi_2(\eta_t).$$

Notice that $a(t)=\alpha_t L\|\bar{v}(t)-\bar{y}(t)\|+2\lambda L\eta_t\|g(t)\|\leq L\|\bar{x}(t)-\bar{y}(t)\|+\frac{8}{1-\sigma}L\eta_t\|g(t)\|$. The statement of the lemma follows. □

C. Proof of Theorem 2

We first introduce Lemma 13 regarding the asymptotic behavior of α_t and λ_t. The proof can be found in Appendix-E of [32].

Lemma 13. *When the vanishing step size is used ($\eta_t=\frac{\eta}{(t+t_0)^\beta}$, $t_0\geq 1$, $\beta\in(0,2)$), and $\eta_0<\frac{1}{4L}$ (equivalently $\alpha_0<\frac{1}{2}$), we have*

(i) $\alpha_t\leq\frac{2}{t+1}$.

(ii) $\lambda_t=O(\frac{1}{t^{2-\beta}})$.

(iii) $\lambda_t\geq\frac{D(\beta,t_0)}{(t+t_0)^{2-\beta}}$ *where $D(\beta,t_0)$ is some constant that only depends on β and t_0, given by* $D(\beta,t_0)=\frac{1}{(t_0+3)^2 e^{16+\frac{6}{2-\beta}}}$.

Now we proceed to prove Theorem 2.

Proof of Theorem 2: It is easy to check that all the conditions of Lemma 6 and 13 are satisfied, hence the conclusions of Lemma 6 and 13 hold. The major step of proving the theorem is to show the following inequality,

$$\lambda_t(\Phi_0(x^*)-f^*)+\phi_t^*\geq f(\bar{x}(t)).\tag{22}$$

If (22) is true, by (22) and (10), we have

$$\begin{aligned}
f(\bar{x}(t))&\leq\phi_t^*+\lambda_t(\Phi_0(x^*)-f^*)\\
&\leq\Phi_t(x^*)+\lambda_t(\Phi_0(x^*)-f^*)\\
&\leq f^*+2\lambda_t(\Phi_0(x^*)-f^*).
\end{aligned}$$

Hence $f(\bar{x}(t))-f^*=O(\lambda_t)=O(\frac{1}{t^{2-\beta}})$, i.e. the desired result of the theorem follows.

Now we use induction to prove (22). Firstly, (22) is true for $t=0$, since $\phi_0^*=f(\bar{x}(0))$ and $\Phi_0(x^*)>f^*$. Suppose it's true for $0,1,2,\ldots,t$. For $0\leq k\leq t$, by (10), $\Phi_k(x^*)\leq f^*+\lambda_k(\Phi_0(x^*)-f^*)$. Hence

$$\phi_k^*+\frac{\gamma_k}{2}\|x^*-\bar{v}(k)\|^2\leq f^*+\lambda_k(\Phi_0(x^*)-f^*).$$

Using the induction assumption, we get

$$f(\bar{x}(k))+\frac{\gamma_k}{2}\|x^*-\bar{v}(k)\|^2\leq f^*+2\lambda_k(\Phi_0(x^*)-f^*).\tag{23}$$

Since $f(\bar{x}(k))\geq f^*$ and $\gamma_k=\lambda_k\gamma_0$, we have $\|x^*-\bar{v}(k)\|^2\leq\frac{4}{\gamma_0}(\Phi_0(x^*)-f^*)$. Since $\bar{v}(k)=\frac{1}{\alpha_k}(\bar{y}(k)-\bar{x}(k))+\bar{x}(k)$, we have $\|\bar{v}(k)-x^*\|^2=\|\frac{1}{\alpha_k}(\bar{y}(k)-\bar{x}(k))+\bar{x}(k)-x^*\|^2\geq\frac{1}{2\alpha_k^2}\|\bar{y}(k)-\bar{x}(k)\|^2-\|\bar{x}(k)-x^*\|^2$. By (23), $f(\bar{x}(k))\leq 2\Phi_0(x^*)-f^*=2f(\bar{x}(0))-f^*+\gamma_0\|\bar{v}(0)-x^*\|^2$. Also since $\gamma_0=\frac{L}{1-\alpha_0}<2L$, we have $\bar{x}(k)$ lies within the $(2f(\bar{x}(0))-f^*+2L\|\bar{v}(0)-x^*\|^2)$-level set of f. By Assumption 3 and Proposition B.9 of [31], we have the level set is compact. Hence we have $\|\bar{x}(k)-x^*\|\leq R$ where R is the diameter of that level set. Combining the above arguments, we get

$$\begin{aligned}
&\|\bar{y}(k)-\bar{x}(k)\|^2\\
&\leq 2\alpha_k^2(\|\bar{v}(k)-x^*\|^2+\|\bar{x}(k)-x^*\|^2)\\
&\leq 2\alpha_k^2[R^2+\frac{4}{\gamma_0}(f(\bar{x}(0))-f^*)+2\|\bar{v}(0)-x^*\|^2]\\
&\leq 2\alpha_k^2\underbrace{[R^2+4\|\bar{v}(0)-x^*\|^2]}_{\triangleq C_1}
\end{aligned}\tag{24}$$

where C_1 is a constant that does *not* depend on η.

Next, we consider (12),

$$\begin{aligned}
&\phi_{t+1}^*-f(\bar{x}(t+1))\\
&=(1-\alpha_t)(\phi_t^*-f(\bar{x}(t)))+(1-\alpha_t)f(\bar{x}(t))+\alpha_t\hat{f}(t)\\
&\quad-\frac{1}{2}\eta_t\|g(t)\|^2+\alpha_t\langle g(t),\bar{v}(t)-\bar{y}(t)\rangle-f(\bar{x}(t+1))\\
&\overset{(a)}{\geq}(1-\alpha_t)(\phi_t^*-f(\bar{x}(t)))+\alpha_t\hat{f}(t)-\frac{1}{2}\eta_t\|g(t)\|^2\\
&\quad+(1-\alpha_t)\{\hat{f}(t)+\langle g(t),\bar{x}(t)-\bar{y}(t)\rangle\}\\
&\quad+\alpha_t\langle g(t),\bar{v}(t)-\bar{y}(t)\rangle-f(\bar{x}(t+1))\\
&\overset{(b)}{=}(1-\alpha_t)(\phi_t^*-f(\bar{x}(t)))+\hat{f}(t)
\end{aligned}$$

$$- \frac{1}{2}\eta_t\|g(t)\|^2 - f(\bar{x}(t+1)) \tag{25}$$

where (a) is due to Lemma 5 and (b) is due to $\alpha_t(\bar{v}(t) - \bar{y}(t)) + (1-\alpha_t)(\bar{x}(t) - \bar{y}(t)) = 0$. By Lemma 5 and Lemma 6,

$$\begin{aligned}
&f(\bar{x}(t+1)) \\
&\le \hat{f}(t) - (\eta_t - L\eta_t^2)\|g(t)\|^2 + 2L\kappa^2\chi_2(\eta_t)^2 \\
&\quad \times (L^2\|\bar{x}(t) - \bar{y}(t)\|^2 + \frac{64}{(1-\sigma)^2}L^2\eta_t^2\|g(t)\|^2).
\end{aligned} \tag{26}$$

Combining the above with (25), we get,

$$\begin{aligned}
&\phi_{t+1}^* - f(\bar{x}(t+1)) \\
&\ge (1-\alpha_t)(\phi_t^* - f(\bar{x}(t))) \\
&\quad + (\frac{1}{2}\eta_t - L\eta_t^2 - \frac{4608L^3\chi_2(\eta_t)^2\eta_t^2}{(1-\sigma)^4})\|g(t)\|^2 \\
&\quad - 2\kappa^2\chi_2(\eta_t)^2L^3\|\bar{x}(t) - \bar{y}(t)\|^2 \\
&\ge (1-\alpha_t)(\phi_t^* - f(\bar{x}(t))) - 2\kappa^2\chi_2(\eta_t)^2L^3\|\bar{x}(t) - \bar{y}(t)\|^2
\end{aligned} \tag{27}$$

where we have used the fact that by step size condition (ii),

$$\begin{aligned}
&\frac{1}{2}\eta_t - L\eta_t^2 - \frac{4608L^3\chi_2(\eta_t)^2\eta_t^2}{(1-\sigma)^4} \\
&\ge \frac{1}{2}\eta_t - L\eta_t^2 - \frac{4608L^3\eta_t^2}{(1-\sigma)^4}\frac{4\eta_t^{2/3}}{L^{4/3}} \\
&\ge \eta_t(1/2 - \frac{18433L\eta}{(1-\sigma)^4}) > 0.
\end{aligned}$$

Hence, expanding (27) recursively, we get

$$\begin{aligned}
&\phi_{t+1}^* - f(\bar{x}(t+1)) \\
&\ge -\sum_{k=0}^{t} 2\kappa^2\chi_2(\eta_k)^2L^3\|\bar{x}(k) - \bar{y}(k)\|^2 \prod_{\ell=k+1}^{t}(1-\alpha_\ell).
\end{aligned}$$

Therefore to finish the induction, we need to show

$$\begin{aligned}
&\sum_{k=0}^{t} 2\kappa^2\chi_2(\eta_k)^2L^3\|\bar{x}(k) - \bar{y}(k)\|^2 \prod_{\ell=k+1}^{t}(1-\alpha_\ell) \\
&\le (\Phi_0(x^*) - f^*)\lambda_{t+1}.
\end{aligned}$$

Notice that

$$\begin{aligned}
&\sum_{k=0}^{t} \frac{2\kappa^2\chi_2(\eta_k)^2L^3\|\bar{x}(k) - \bar{y}(k)\|^2\prod_{\ell=k+1}^{t}(1-\alpha_\ell)}{(\Phi_0(x^*) - f^*)\lambda_{t+1}} \\
&\overset{(a)}{\le} \sum_{k=0}^{t} \frac{4\kappa^2L^3}{L\|\bar{v}(0) - x^*\|^2}\chi_2(\eta_k)^2\|\bar{x}(k) - \bar{y}(k)\|^2\frac{1}{\lambda_{k+1}} \\
&\overset{(b)}{\le} \sum_{k=0}^{t} \frac{4(6/(1-\sigma))^2L^2}{\|\bar{v}(0) - x^*\|^2}(\frac{2}{L^{2/3}}\eta_k^{1/3})^2 2C_1\alpha_k^2\frac{1}{\lambda_{k+1}} \\
&= \sum_{k=0}^{t} \underbrace{\frac{1152L^{2/3}C_1}{(1-\sigma)^2\|\bar{v}(0) - x^*\|^2}}_{\triangleq C_2}\eta_k^{2/3}\alpha_k^2\frac{1}{\lambda_{k+1}}
\end{aligned}$$

where C_2 is a costant that does *not* depend on η, and in (a) we have used $\Phi_0(x^*) - f^* \ge \frac{L}{2}\|\bar{v}(0) - x^*\|^2 > 0$, and in (b), we have used the bound on $\chi_2(\eta_k)$ (Lemma 6) and the bound on $\|\bar{x}(k) - \bar{y}(k)\|$ (equation (24)). Now by Lemma 13, we get,

$$\begin{aligned}
\sum_{k=0}^{t}\eta_k^{2/3}\alpha_k^2\frac{1}{\lambda_{k+1}} &\le \sum_{k=0}^{t}\frac{\eta^{2/3}}{(k+t_0)^{\frac{2}{3}\beta}}\frac{4}{(k+1)^2}\frac{(k+1+t_0)^{2-\beta}}{D(\beta,t_0)} \\
&\overset{(a)}{\le} \eta^{2/3}\frac{4(t_0+1)^{2-\beta}}{D(\beta,t_0)}\sum_{k=0}^{\infty}\frac{1}{(k+1)^{\frac{5}{3}\beta}} \\
&\overset{(b)}{\le} \eta^{2/3}\frac{4(t_0+1)^{2-\beta}}{D(\beta,t_0)} \times \frac{2}{\beta - 0.6}
\end{aligned}$$

where in (a) we have used, $k + t_0 \ge k + 1$, $k + 1 + t_0 \le (t_0 + 1)(k + 1)$; in (b) we have used $\frac{5}{3}\beta > 1$. So, we have

$$\begin{aligned}
&\sum_{k=0}^{t}\frac{2\kappa^2\chi_2(\eta_k)^2L^3\|\bar{x}(k) - \bar{y}(k)\|^2\prod_{\ell=k+1}^{t}(1-\alpha_\ell)}{(\Phi_0(x^*) - f^*)\lambda_{t+1}} \\
&\le \eta^{2/3}C_2\frac{8(t_0+1)^{2-\beta}}{D(\beta,t_0)(\beta-0.6)} < 1
\end{aligned}$$

where in the last inequality, we have simply required $\eta^{2/3} < \frac{D(\beta,t_0)(\beta-0.6)}{8(t_0+1)^{2-\beta}C_2}$ (i.e. step size condition (iii)), which is possible since the constants C_2 and $D(\beta, t_0)$ do not depend on η. So the induction is complete and we have (22) is true. □

IV. Numerical Experiments

We simulate our algorithm and compare it with other algorithms. We choose $n = 100$ agents and the graph is generated using the Erdos-Renyi model [35] with connectivity probability 0.3. The weight matrix W is chosen using the Laplacian method [16, Sec. 2.4]. We will compare our algorithm Acc-DNGD with Distributed Gradient Descent (DGD) in [6] with a vanishing step size, the "EXTRA" algorithm in [16] (with $\tilde{W} = \frac{W+I}{2}$), the algorithm studied in [19]–[25] (we name it "Acc-DGD"), the "D-NG" method in [28]. We will also compare with two centralized methods that directly optimize f: Centralized Gradient Descent (CGD) and Centralized Nesterov Gradient Descent (CNGD (2)). Each element of the initial point $x_i(0)$ is drawn from i.i.d. Gaussian with mean 0 and variance 25. The objective functions are given by,

$$f_i(x) = \begin{cases} \frac{1}{m}\langle a_i, x\rangle^m + \langle b_i, x\rangle & \text{if } |\langle a_i, x\rangle| \le 1, \\ |\langle a_i, x\rangle| - \frac{m-1}{m} + \langle b_i, x\rangle & \text{if } |\langle a_i, x\rangle| > 1, \end{cases}$$

where $m = 12$, $a_i, b_i \in \mathbb{R}^N$ ($N = 4$) are vectors whose entries are i.i.d. Gaussian with mean 0 and variance 1, with the exception that b_n is set to be $b_n = -\sum_{i=1}^{n-1} b_i$ s.t. $\sum_i b_i = 0$. It is easy to check that f_i is convex and smooth, but not strongly convex (around the minimizer).

The selection of the objective functions is intended to test the sublinear convergence rate $\frac{1}{t^{2-\beta}}$ ($\beta > 0.6$) of our algorithm Acc-DNGD (3) and the conjecture that the $\frac{1}{t^{2-\beta}}$ rate still holds even if $\beta \in [0, 0.6]$ (cf. Theorem 2 and the comments following it). Therefore, we do two runs of our algorithm Acc-DNGD, one with $\beta = 0.61$ and the other with $\beta = 0$. The results are shown in Figure 1, where the x-axis is the iteration t, and the y-axis is the average objective error $\frac{1}{n}\sum f(x_i(t)) - f^*$ for distributed methods, or objective error $f(x(t)) - f^*$ for centralized methods. Notice that Figure 1 is a double log plot. It shows that Acc-DNGD with $\beta = 0.61$ performs faster than $1/t^{1.39}$, while D-NG, CGD and CGD-based distributed methods (DGD, Acc-DGD, EXTRA) are slower than $1/t^{1.39}$. Further, both Acc-DNGD with $\beta = 0$ and CNGD are faster than $\frac{1}{t^2}$.

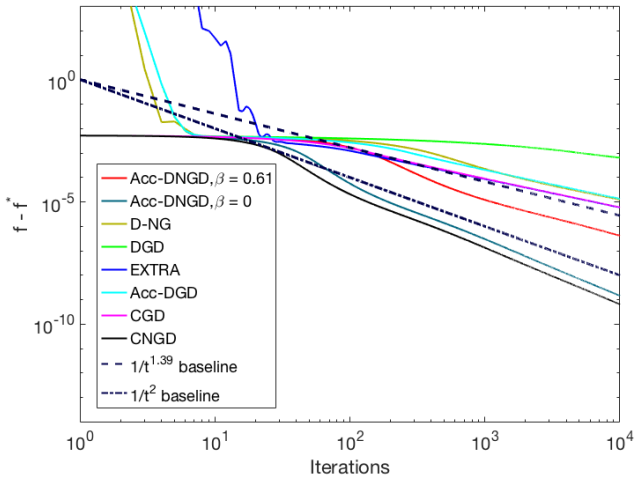

Fig. 1: Simulation results. Steps sizes: Acc-DNGD with $\beta = 0.61$: $\eta_t = \frac{0.0045}{(t+1)^{0.61}}$, $\alpha_0 = 0.7071$; Acc-DNGD with $\beta = 0$: $\eta_t = 0.0045$, $\alpha_0 = 0.7071$; D-NG: $\eta_t = \frac{0.0091}{t+1}$; DGD: $\eta_t = \frac{0.0091}{\sqrt{t}}$; EXTRA: $\eta = 0.0091$; Acc-DGD: $\eta = 0.0045$; CGD: $\eta = 0.0091$; CNGD: $\eta = 0.0091$, $\alpha_0 = 0.5$.

V. Conclusion

In this paper we propose an Accelerated Distributed Nesterov Gradient Descent algorithm for distributed optimization of convex and smooth functions. We show a general $O(\frac{1}{t^{1.4-\epsilon}})$ $(\forall \epsilon \in (0, 1.4))$ convergence rate, and an improved $O(\frac{1}{t^2})$ convergence rate when the objective functions satisfy an additional property. Future work includes giving tighter analysis of the convergence rates.

References

[1] B. Johansson, "On distributed optimization in networked systems," 2008.

[2] J. A. Bazerque and G. B. Giannakis, "Distributed spectrum sensing for cognitive radio networks by exploiting sparsity," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1847–1862, 2010.

[3] P. A. Forero, A. Cano, and G. B. Giannakis, "Consensus-based distributed support vector machines," *Journal of Machine Learning Research*, vol. 11, no. May, pp. 1663–1707, 2010.

[4] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," in *1984 American Control Conference*, 1984, pp. 484–489.

[5] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

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

[7] I. Lobel and A. Ozdaglar, "Convergence analysis of distributed subgradient methods over random networks," in *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*. IEEE, 2008, pp. 353–360.

[8] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: convergence analysis and network scaling," *Automatic control, IEEE Transactions on*, vol. 57, no. 3, pp. 592–606, 2012.

[9] S. S. Ram, A. Nedić, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *Journal of optimization theory and applications*, vol. 147, no. 3, pp. 516–545, 2010.

[10] A. Nedic and A. Olshevsky, "Stochastic gradient-push for strongly convex functions on time-varying directed graphs," *arXiv preprint arXiv:1406.2075*, 2014.

[11] ——, "Distributed optimization over time-varying directed graphs," *Automatic Control, IEEE Transactions on*, vol. 60, no. 3, pp. 601–615, 2015.

[12] I. Matei and J. S. Baras, "Performance evaluation of the consensus-based distributed subgradient method under random communication topologies," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 4, pp. 754–771, 2011.

[13] A. Olshevsky, "Linear time average consensus on fixed graphs and implications for decentralized optimization and multi-agent control," *arXiv preprint arXiv:1411.4186*, 2014.

[14] M. Zhu and S. Martínez, "On distributed convex optimization under inequality and equality constraints," *Automatic Control, IEEE Transactions on*, vol. 57, no. 1, pp. 151–164, 2012.

[15] I. Lobel, A. Ozdaglar, and D. Feijer, "Distributed multi-agent optimization with state-dependent communication," *Mathematical Programming*, vol. 129, no. 2, pp. 255–284, 2011.

[16] 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.

[17] C. Xi and U. A. Khan, "On the linear convergence of distributed optimization over directed graphs," *arXiv preprint arXiv:1510.02149*, 2015.

[18] J. Zeng and W. Yin, "Extrapush for convex smooth decentralized optimization over directed networks," *arXiv preprint arXiv:1511.02942*, 2015.

[19] J. Xu, S. Zhu, Y. C. Soh, and L. Xie, "Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes," in *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 2055–2060.

[20] P. Di Lorenzo and G. Scutari, "Distributed nonconvex optimization over networks," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2015 IEEE 6th International Workshop on*. IEEE, 2015, pp. 229–232.

[21] P. Di Lorenzo and G. Scutari, "Next: In-network nonconvex optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 2, pp. 120–136, 2016.

[22] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *arXiv preprint arXiv:1605.07112*, 2016.

[23] A. Nedich, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *arXiv preprint arXiv:1607.03218*, 2016.

[24] A. Nedic, A. Olshevsky, W. Shi, and C. A. Uribe, "Geometrically convergent distributed optimization with uncoordinated step-sizes," *arXiv preprint arXiv:1609.05877*, 2016.

[25] C. Xi and U. A. Khan, "Add-opt: Accelerated distributed directed optimization," *arXiv preprint arXiv:1607.04757*, 2016.

[26] Y. Nesterov, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2013, vol. 87.

[27] G. Qu and N. Li, "Accelerated distributed nesterov gradient descent for smooth and strongly convex functions," in *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*. IEEE, 2016, pp. 209–216.

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

[29] A. Olshevsky and J. N. Tsitsiklis, "Convergence speed in distributed consensus and averaging," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 33–55, 2009.

[30] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, Jan 2007.

[31] D. P. Bertsekas, "Nonlinear programming," 1999.

[32] N. L. Guannan Qu. (2017) Accelerated distributed nesterov gradient descent for convex and smooth functions. [Online]. Available: http://scholar.harvard.edu/files/gqu/files/cdc2017fullversion.pdf

[33] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, no. 1-2, pp. 37–75, 2014.

[34] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[35] P. Erdos and A. Renyi, "On random graphs i," *Publ. Math. Debrecen*, vol. 6, pp. 290–297, 1959.