Journal of Machine Learning Research 20 (2019) 1-31 Submitted 6/19; Revised 9/19; Published 9/19

Optimal Convergence Rates for Convex Distributed Optimization in Networks

Kevin Scaman KEVIN.SCAMAN@HUAWEI.COM
Huawei Noah's Ark Lab, Paris, France

Francis Bach FRANCIS.BACH@INRIA.FR
INRIA, Ecole Normale Supérieure, PSL Research University, Paris, France

Sébastien Bubeck SEBUBECK@MICROSOFT.COM
Microsoft Research, Redmond, United States

Yin Tat Lee YILE@MICROSOFT.COM
University of Washington, Seattle, United States

Laurent Massoulié LAURENT.MASSOULIE@INRIA.FR
MSR-INRIA Joint Center, Paris, France

Editor: Kilian Weinberger

Abstract

This work proposes a theoretical analysis of distributed optimization of convex functions using a network of computing units. We investigate this problem under two communication schemes (centralized and decentralized) and four classical regularity assumptions: Lipschitz continuity, strong convexity, smoothness, and a combination of strong convexity and smoothness. Under the decentralized communication scheme, we provide matching upper and lower bounds of complexity along with algorithms achieving this rate up to logarithmic constants. For non-smooth objective functions, while the dominant term of the error is in $O(1/\sqrt{t})$, the structure of the communication network only impacts a second-order term in $O(1/t)$, where t is time. In other words, the error due to limits in communication resources decreases at a fast rate even in the case of non-strongly convex objective functions. Such a convergence rate is achieved by the novel *multi-step primal-dual* (MSPD) algorithm. Under the centralized communication scheme, we show that the naive distribution of standard optimization algorithms is optimal for smooth objective functions, and provide a simple yet efficient algorithm called *distributed randomized smoothing* (DRS) based on a local smoothing of the objective function for non-smooth functions. We then show that DRS is within a $d^{1/4}$ multiplicative factor of the optimal convergence rate, where d is the underlying dimension.

Keywords: distributed optimization, convex optimization, first-order methods

1. Introduction

Distributed optimization finds many applications in machine learning, for example when the data set is large and training is achieved using a cluster of computing units. As a result, many algorithms were recently introduced to minimize the average $\bar{f} = \frac{1}{n}\sum_{i=1}^{n} f_i$ of functions f_i, $i = 1, \ldots, n$, which are respectively accessible by separate nodes in a network (Nedic and Ozdaglar, 2009; Boyd et al., 2011; Duchi et al., 2012a; Shi et al., 2015). Most often, these algorithms alternate between local and incremental improvement steps (such as gradient steps) with communication steps between nodes in the network, and come with a variety of convergence rates (see, for example, Shi et al., 2014, 2015; Jakovetić et al., 2015; Nedic et al., 2017).

The main contribution of this paper is to provide optimal convergence rates and their corresponding optimal algorithms for distributed convex problems under a large panel of regularity assumptions and communication schemes. More specifically, we consider two communication schemes and four regularity assumptions. Communication is either achieved through a *master/slave* approach that we refer to as *centralized*, or in a *decentralized* scheme in which communication is performed using the *gossip* algorithm (Boyd et al., 2006). Moreover, the four regularity assumptions that we investigate in this paper are: Lipschitz continuity, strong convexity, smoothness, and both strong convexity and smoothness. These settings are summarized in Table 1 along with their corresponding upper and lower complexity bounds. These bounds depend on the time τ necessary to communicate a vector between two neighbors in the communication graph, the diameter of the communication graph Δ, its *mixing time* $\widetilde{\Delta}$ (see Section 4), the dimension of the optimization problem d and distance to optimum R, and the Lipschitz, strong convexity, smoothness constants, and condition number of the objective function (L_g, α_g, β_g and κ_g, respectively, for the *global* objective function, and L_ℓ, α_ℓ, β_ℓ and κ_ℓ for an aggregation of the *local* individual functions of each computing unit, see Section 2).

Under decentralized communication, we provide in Section 4 matching upper and lower bounds of complexity.[1] Moreover, we propose the first optimal algorithm for non-smooth decentralized optimization, called *multi-step primal-dual* (MSPD). Under centralized communication, we show in Section 3 that, for smooth objectives, a naïve distribution of accelerated gradient descent is optimal, while for non-smooth objectives, distributing the simple smoothing approach introduced by Duchi et al. (2012b) yields fast convergence rates with respect to communication. This algorithm, called *distributed randomized smoothing* (DRS), achieves a convergence rate matching the lower bound up to a $d^{1/4}$ multiplicative factor, where d is the dimensionality of the problem.

The analysis of non-smooth optimization differs from the smooth setting in two major aspects: (1) the naïve *master/slave* distributed algorithm is in this case not optimal, and (2) the convergence rates differ between communication and local computations. More specifically, errors due to limits in communication resources enjoy fast convergence rates, as we establish by formulating the optimization problem as a composite saddle-point problem with a smooth term for communication and non-smooth term for the optimization of the local functions (see Section 4 and Equation 11 for more details).

1.1. Related Work

Many algorithms were proposed to solve the decentralized optimization of an average of functions, from early decentralized algorithms (Nedic and Ozdaglar, 2009; Boyd et al., 2011; Duchi et al., 2012a; Wei and Ozdaglar, 2012; Jakovetić et al., 2014; Shi et al., 2014, 2015;

[1] The lower bounds for decentralized communication for strongly convex, smooth, and both strongly convex and smooth settings, are met up to logarithmic factors using the PSTM and R-Sliding algorithms of Dvinskikh and Gasnikov (2019).

Setting	**Lower bound**	**Upper bound**	**Algorithm**
Lipschitz continuity + convexity			
Centr.	$\Omega\left(\left(\frac{L_gR}{\varepsilon}\right)^2 + \frac{L_gR}{\varepsilon}\Delta\tau\right)$	$O\left(\left(\frac{L_gR}{\varepsilon}\right)^2 + \frac{L_gR}{\varepsilon}\Delta\tau d^{1/4}\right)$	DRS (Alg. 1)
Decentr.	$\Omega\left(\left(\frac{L_\ell R}{\varepsilon}\right)^2 + \frac{L_\ell R}{\varepsilon}\widetilde{\Delta}\tau\right)$	matching lower bound	MSPD (Alg. 3)
Lipschitz continuity + strong convexity			
Centr.	$\Omega\left(\frac{L_g^2}{\alpha_g\varepsilon} + \sqrt{\frac{L_g^2}{\alpha_g\varepsilon}}\Delta\tau\right)$	$O\left(\frac{L_g^2}{\alpha_g\varepsilon} + \sqrt{\frac{L_g^2}{\alpha_g\varepsilon}}\Delta\tau d^{1/4}\log\left(\frac{1}{\varepsilon}\right)\right)$	DRS (Alg. 2)
Decentr.	$\Omega\left(\frac{L_\ell^2}{\alpha_\ell\varepsilon} + \sqrt{\frac{L_\ell^2}{\alpha_\ell\varepsilon}}\widetilde{\Delta}\tau\right)$	matching lower bound[1]	R-Sliding[1]
Smoothness + convexity			
Centr.	$\Omega\left(R\sqrt{\frac{\beta_g}{\varepsilon}}(1+\Delta\tau)\right)$	matching lower bound	AGD
Decentr.	$\Omega\left(R\sqrt{\frac{\beta_\ell}{\varepsilon}}(1+\widetilde{\Delta}\tau)\right)$	matching lower bound[1]	PSTM[1]
Smoothness + strong convexity			
Centr.	$\Omega\left(\sqrt{\kappa_g}\ln\left(\frac{1}{\varepsilon}\right)(1+\Delta\tau)\right)$	matching lower bound	AGD
Decentr.	$\Omega\left(\sqrt{\kappa_\ell}\ln\left(\frac{1}{\varepsilon}\right)(1+\widetilde{\Delta}\tau)\right)$	matching lower bound[1]	PSTM[1]

Table 1: Upper and lower bounds on the minimax optimal time needed to reach an $\varepsilon > 0$ precision for four regularization assumptions and two communication settings, along with corresponding optimal algorithms (AGD stands for the naïve distribution of accelerated gradient descent, see Section 3 for more details).

Mokhtari and Ribeiro, 2016) to more recent improvements (Qu and Li, 2016; Jiang et al., 2017; He et al., 2018; Pu et al., 2018; Uribe et al., 2018; Koloskova et al., 2019; Xin et al., 2019; Jakovetić, 2019). Another line of work considers more restrictive assumptions on the optimization problems to obtain fast convergence rates (Lee et al., 2017; Jesús Arroyo, 2016; Tian and Gu, 2017). A sheer amount of work was devoted to improving the convergence rate of these algorithms (Shi et al., 2014; Jakovetić et al., 2015). In the case of non-smooth optimization, fast communication schemes were developed by Lan et al. (2017); Jaggi et al. (2014), although precise optimal convergence rates were not obtained. Our decentralized algorithm is closely related to the recent primal-dual algorithm of Lan et al. (2017) which enjoys fast communication rates in a decentralized and stochastic setting. Unfortunately, their algorithm lacks gossip acceleration to reach optimality with respect to communication time. Finally, optimal convergence rates for distributed algorithms were investigated by Scaman et al. (2017) for smooth and strongly convex objective functions, Shamir (2014); Arjevani and Shamir (2015) for totally connected networks, and more recently by Dvinskikh and Gasnikov (2019) who derive optimal algorithms (up to logarithmic multiplicative factors) for decentralized optimization, yet without the corresponding complexity lower bounds. Our analysis proves that these algorithms are indeed, up to logarithmic multiplicative factors, optimal in a minimax sense. A preliminary version of this work with only the analysis for the Lipschitz-continuous case (first line in Table 1) is available as a conference paper (Scaman et al., 2018).

2. Distributed Optimization Setting

In this section, we provide a detailed presentation of the distributed optimization setting considered in this work. The definitions are analogous to that of Scaman et al. (2017).

2.1. Optimization Problem

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a connected simple (i.e., undirected) graph of n computing units and diameter Δ, each having access to a convex function f_i over a convex set $\mathcal{K} \subset \mathbb{R}^d$. We consider minimizing the average of the local functions

$$\min_{\theta \in \mathcal{K}} \ \bar{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} f_i(\theta) \,, \tag{1}$$

in a distributed setting. More specifically, we assume that each computing unit can compute a subgradient $\nabla f_i(\theta)$ of its own function in one unit of time, and communicate values (i.e., vectors in $\mathbb{R}^d$) to its neighbors in $\mathcal{G}$. A *direct* communication along the edge $(i,j) \in \mathcal{E}$ requires a time $\tau \geq 0$. These actions may be performed asynchronously and in parallel, and each machine i possesses a local version of the parameter, which we refer to as $\theta_i \in \mathcal{K}$.

2.2. Regularity Assumptions

Optimal convergence rates depend on the precise set of assumptions applied to the objective function. In our case, we will consider four different regularity assumptions:

1. **Convexity:** a function f is convex if, for all $\theta, \theta' \in \mathcal{K}$ and $a \in [0,1]$,

$$f(a\theta + (1-a)\theta') \leq a f(\theta) + (1-a) f(\theta') \,.$$

2. **Strong convexity:** a function f is α-strongly convex if $f(\theta) - \frac{\alpha}{2}\|\theta\|_2^2$ is convex.

3. **Lipschitz continuity:** a function f is L-Lipschitz continuous if, for all $\theta, \theta' \in \mathcal{K}$,

$$|f(\theta) - f(\theta')| \leq L \|\theta - \theta'\|_2 \,.$$

4. **Smoothness:** a function f is β-smooth if its gradient ∇f is β-Lipschitz continuous.

Moreover, when a function is both α-strongly convex and β-smooth, we denote by $\kappa = \beta/\alpha$ its condition number. In our analysis, we will denote by L_g, α_g, β_g and κ_g the characteristics of the (global) function $\bar{f}$, while, for $i \in [\![1, n]\!]$, we will refer to L_i, α_i, β_i and κ_i for characteristics of each (local) function f_i. Finally, for the sake of clarity, we will aggregate these local characteristics into the following values: $L_\ell = \sqrt{\frac{1}{n}\sum_{i=1}^n L_i^2}$, $\alpha_\ell = \min_i \alpha_i$, $\beta_\ell = \max_i \beta_i$, and $\kappa_\ell = \beta_\ell/\alpha_\ell$.

Global regularity is always *weaker* than local regularity, as we always have $L_g \leq L_\ell$, $\alpha_g \geq \alpha_\ell$, $\beta_g \leq \beta_\ell$ and $\kappa_g \leq \kappa_\ell$. Moreover, we may have $L_g = 0$ and L_ℓ arbitrarily large, for example with two linear functions $f_1(x) = -f_2(x) = ax$ and $a \to +\infty$. We will see in the following sections that the local regularity assumption is easier to analyze and leads to matching upper and lower bounds. For the global regularity assumption, we only provide algorithms with a $d^{1/4}$ competitive ratio, where d is the dimension of the problem. Finding an optimal distributed algorithm for global regularity is, to our understanding, a much more challenging task and is left for future work.

Finally, we assume that the feasible region $\mathcal{K}$ is convex and bounded, and denote by R the radius of a ball containing $\mathcal{K}$, i.e.,

$$\forall \theta \in \mathcal{K}, \ \ \|\theta - \theta_0\|_2 \leq R\,,$$

where $\theta_0 \in \mathcal{K}$ is the initial value of the algorithm, that we set to $\theta_0 = 0$ without loss of generality.

2.3. Black-box Optimization Procedure

The lower bounds provided hereafter depend on the notion of black-box optimization procedures for the problem in Equation 1. The definition is analogous to that of Scaman et al. (2017), except that gradients of the Fenchel conjugate of local functions are considered too expensive to compute. This allows us to focus on primal methods which are generally easier to apply (less assumptions required on the objective function) and faster to compute (the Fenchel conjugate is, except for very specific functions, hard to approximate). A black-box optimization procedure is a distributed algorithm verifying the following constraints:

1. **Local memory:** each node i can store past values in a (finite) internal memory $\mathcal{M}_{i,t} \subset \mathbb{R}^d$ at time $t \geq 0$. These values can be accessed and used at time t by the algorithm run by node i, and are updated either by local computation or by communication (defined below), that is, for all $i \in [\![1, n]\!]$,

$$\mathcal{M}_{i,t} \subset \mathcal{M}_{i,t}^{\text{comp}} \cup \mathcal{M}_{i,t}^{\text{comm}}.$$

2. **Local computation:** each node i can, at time t, compute a subgradient of its local function $\nabla f_i(\theta)$ for a value $\theta \in \mathcal{M}_{i,t-1}$ in the node's internal memory before the computation.

$$\mathcal{M}_{i,t}^{\text{comp}} = \mathrm{Span}\left(\{\theta, \nabla f_i(\theta) : \theta \in \mathcal{M}_{i,t-1}\}\right).$$

3. **Local communication:** each node i can, at time t, share a value to all or part of its neighbors, that is, for all $i \in [\![1, n]\!]$,

$$\mathcal{M}_{i,t}^{\text{comm}} = \mathrm{Span}\Bigg(\bigcup_{(j,i)\in\mathcal{E}} \mathcal{M}_{j,t-\tau}\Bigg).$$

4. **Output value:** each node i must, at time t, specify one vector in its memory as local output of the algorithm, that is, for all $i \in [\![1, n]\!]$,

$$\theta_{i,t} \in \mathcal{M}_{i,t} \,.$$

Hence, a black-box procedure will return n output values—one for each computing unit—and our analysis will focus on ensuring that *all local output values* are converging to the optimal parameter of Equation 1. For simplicity, we assume that all nodes start with the simple internal memory $\mathcal{M}_{i,0} = \{0\}$. Note that communications and local computations may be performed in parallel and asynchronously.

3. Centralized Optimization under Global Regularity

The most standard approach for distributing a first-order optimization method consists in computing a gradient (or subgradient when the objective function is non-smooth) of the average function

$$\nabla \bar{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(\theta) \,,$$

where $\nabla f_i(\theta)$ is a gradient (or any subgradient of f_i at θ), by sending the current parameter θ_t to all nodes, performing the computation of all local subgradients in parallel and averaging them on a master node. Since each iteration requires communicating twice to the whole network (once for θ_t and once for sending the local subgradients to the master node, which both take a time $\Delta\tau$ where Δ is the diameter of the network) and one gradient computation (on each node and performed in parallel), the time to reach a precision ε with such a distributed optimization algorithm is upper-bounded by

$$O\bigg(g(\varepsilon)(1 + \Delta\tau)\bigg), \tag{2}$$

where $g(\varepsilon)$ is the number of iterations required for the single-machine optimization algorithm to achieve a precision ε. Note that, as the single-machine optimization algorithm is directly used for $\bar{f}$, this convergence rate depends on the global characteristics of objective function.

For smooth objective functions, we will see in Section 3.2 that distributing in such a way Nesterov's accelerated gradient descent leads to an optimal convergence rate. For non-smooth objective functions, however, the simple scheme is not optimal: the number of subgradient computations in Equation 2 (i.e., the term not proportional to τ) cannot be improved, since it is already optimal for objective functions defined on only one machine—see for example Theorem 3.13 p. 280 from Bubeck (2015). However, quite surprisingly, the error due to communication time may benefit from fast convergence rates in $O(RL_g/\varepsilon)$. This result is already known under the local regularity assumption (i.e., replacing L_g with L_ℓ or even $\max_i L_i$) in the case of decentralized optimization (Lan et al., 2017) or distributed optimization on a totally connected network (Arjevani and Shamir, 2015). To our knowledge, the case of global regularity has not been investigated by prior work.

We first describe algorithms in Section 3.1, and then study their optimality according in Section 3.2.

3.1. Fast Communication Rates for Non-Smooth Objective Functions

We now show that the simple smoothing approach introduced by Duchi et al. (2012b) can lead to fast rates for error due to communication time. Let $\gamma > 0$ and $f : \mathbb{R}^d \to \mathbb{R}$ be a real function. We denote as *smoothed version of* f the following function:

$$f^\gamma(\theta) = \mathbb{E}\left[f(\theta + \gamma X)\right], \tag{3}$$

where $X \sim \mathcal{N}(0, I)$ is a standard Gaussian random variable. The following lemma shows that f^γ is both smooth and a good approximation of f.

Lemma 1 (Lemma $E.3$ of Duchi et al., 2012b) *If $\gamma > 0$, then f^γ is $(\frac{L_g}{\gamma})$-smooth and, for all $\theta \in \mathbb{R}^d$,*

$$f(\theta) \quad\leq\quad f^\gamma(\theta) \quad\leq\quad f(\theta) + \gamma L_g \sqrt{d}\,.$$

Hence, smoothing the objective function allows the use of accelerated optimization algorithms and provides faster convergence rates. Of course, the price to pay is that each computation of the smoothed gradient $\nabla \bar{f}^\gamma(\theta) = \frac{1}{n}\sum_{i=1}^n \nabla f_i^\gamma(\theta)$ now requires, at each iteration m, to sample a sufficient amount of subgradients $\nabla f_i(\theta + \gamma X_{m,k})$ to approximate Equation 3, where $X_{m,k}$ are K i.i.d. Gaussian random variables. At first glance, this algorithm requires all computing units to synchronize on the choice of $X_{m,k}$, which would require to send to all nodes each $X_{m,k}$ and thus incur a communication cost proportional to the number of samples. Fortunately, computing units only need to share one random seed $s \in \mathbb{R}$ and then use a random number generator initialized with the provided seed to generate the same random variables $X_{m,k}$ without the need to communicate any vector. While the theoretical results assume that the randomness is shared, in practice this is done approximately by sharing random seeds and using pseudo-random generators. The overall algorithm, denoted *distributed randomized smoothing* (DRS), uses the randomized smoothing optimization algorithm of Duchi et al. (2012b) adapted to a distributed setting, and is summarized in Alg. 1. The computation of a spanning tree $\mathcal{T}$ in step 1 allows efficient communication to the whole network in time at most $\Delta\tau$. Most of the algorithm (i.e., steps $2, 4, 6, 7, 9, 10$ and 11) are performed on the root of the spanning subtree $\mathcal{T}$, while the rest of the computing units are responsible for computing the smoothed gradient (step 8). The seed s of step 2 is used to ensure that every $X_{m,k}$, although random, is the *same on every node*. Finally, step 10 is a simple orthogonal projection of the gradient step on the convex set $\mathcal{K}$.

3.1.1. Convex Case

We now show that the DRS algorithm converges to the optimal parameter under global Lipschitz regularity.

Theorem 2 *Under global Lipschitz-continuity, Alg. 1 achieves an approximation error $\mathbb{E}\left[\bar{f}(\theta_T)\right] - \bar{f}(\theta^*)$ of at most $\varepsilon > 0$ in a time T_ε upper-bounded by*

$$O\bigg(\Big(\frac{RL_g}{\varepsilon}\Big)^2 + \frac{RL_g}{\varepsilon}(1 + \Delta\tau)d^{1/4}\bigg)\,. \tag{4}$$

Algorithm 1 distributed randomized smoothing (convex case)

```
Input: approximation error $\varepsilon > 0$, communication graph $\mathcal{G}$, $\alpha_0 = 1$, $\alpha_{t+1} = 2/(1 + \sqrt{1 + 4/\alpha_t^2})$
    $T = \left\lceil \frac{20RL_g d^{1/4}}{\varepsilon} \right\rceil$, $K = \left\lceil \frac{5RL_g d^{-1/4}}{\varepsilon} \right\rceil$, $\gamma_t = Rd^{-1/4}\alpha_t$, $\eta_t = \frac{R\alpha_t}{2L_g(d^{1/4} + \sqrt{\frac{t+1}{K}})}$.
Output: optimizer $\theta_T$
Compute a spanning tree $\mathcal{T}$ on $\mathcal{G}$.
Send a random seed $s$ to every node in $\mathcal{T}$.
Initialize the random number generator of each node using $s$.
$x_0 = 0$, $z_0 = 0$, $G_0 = 0$
for $t = 0$ to $T - 1$ do
    $y_t = (1 - \alpha_t)x_t + \alpha_t z_t$
    Send $y_t$ to every node in $\mathcal{T}$.
    Each node $i$ computes $g_i = \frac{1}{K}\sum_{k=1}^K \nabla f_i(y_t + \gamma_t X_{t,k})$, where $X_{t,k} \sim \mathcal{N}(0, I)$
    $G_{t+1} = G_t + \frac{1}{n\alpha_t}\sum_i g_i$
    $z_{t+1} = \operatorname{argmin}_{x \in \mathcal{K}} \|x + \eta_{t+1}G_{t+1}\|_2^2$
    $x_{t+1} = (1 - \alpha_t)x_t + \alpha_t z_{t+1}$
end for
return $\theta_T = x_T$
```

More specifically, Alg. 1 completes its T iterations by time

$$T_\varepsilon \quad \leq \quad 100 \left\lceil \frac{RL_g d^{1/4}}{\varepsilon} \right\rceil \left\lceil \frac{RL_g d^{-1/4}}{\varepsilon} \right\rceil + 40 \left\lceil \frac{RL_g d^{1/4}}{\varepsilon} \right\rceil \Delta\tau \,. \tag{5}$$

Comparing Equation 4 to Equation 2, we can see that our algorithm benefits from faster convergence rates than the naïve distribution of gradient descent (leading to $g(\varepsilon) = (RL_g/\varepsilon)^2$ in Equation 2, which is optimal for single machine optimization of convex non-smooth problems) when the dimension is not too large, and more specifically

$$d \leq \left(\frac{RL_g}{\varepsilon}\right)^4.$$

In practice, this condition is easily met, as $\varepsilon \leq 10^{-2}$ already leads to the condition $d \leq 10^8$ (assuming that the product RL_g is close to 1). Moreover, for problems of moderate dimension, the term $d^{1/4}$ remains a small multiplicative factor (e.g. for $d = 1000$, $d^{1/4} \approx 6$). Finally, note that DRS achieves a linear speedup when communication through the whole network requires a constant time, i.e., $\Delta\tau = O(1)$.

Remark 3 *Several other smoothing methods exist in the literature, notably the* Moreau envelope *(Moreau, 1965) enjoying a dimension-free approximation guarantee. However, the Moreau envelope of an average of functions is difficult to compute (requires a different oracle than computing a subgradient), and unfortunately leads to convergence rates with respect to local Lipschitz characteristics instead of L_g. Moreover, the above analysis can be extended to any rotation invariant probability measure (e.g. uniform on a ball) for the smoothing noise. Such an analysis shows that the convergence rate obtained by using Gaussian random*

Algorithm 2 distributed randomized smoothing (strongly-convex case)

```
Input: approximation error $\varepsilon > 0$, communication graph $\mathcal{G}$,
    $c_{d,\varepsilon} = d^{1/4} \log\left(\frac{8(\bar{f}(\theta_0) - \bar{f}(\theta^*))}{\varepsilon}\right)$, $T = 2\left\lceil \frac{\sqrt{2} L_g c_{d,\varepsilon}}{\sqrt{\alpha_g \varepsilon}} \right\rceil$, $K = \left\lceil \frac{70 L_g c_{d,\varepsilon}^{-1}}{\sqrt{\alpha_g \varepsilon}} \right\rceil$, $\gamma = \frac{\varepsilon}{2 L_g \sqrt{d}}$,
    $n_0 = -1$, $n_1 = \left\lceil \sqrt{\frac{L_g}{\alpha_g \gamma}} \log\left(\frac{4(\bar{f}(\theta_0) - \bar{f}(\theta^*))}{\varepsilon}\right) \right\rceil$, $\forall t > 1$, $n_t = 2^t \left\lceil \sqrt{\frac{L_g}{\alpha_g \gamma}} \log(8) \right\rceil$,
    $\eta_1 = \frac{\gamma}{L_g}$, $\forall t > 1$, $\eta_t = \frac{\gamma}{2^{2t} L_g}$, $\mu_t = \frac{1 - \sqrt{\alpha_g \eta_t}}{1 + \sqrt{\alpha_g \eta_t}}$.
Output: optimizer $\theta_T$
Compute a spanning tree $\mathcal{T}$ on $\mathcal{G}$.
Send a random seed $s$ to every node in $\mathcal{T}$.
Initialize the random number generator of each node using $s$.
for $t = 1$ to $T$ do
    $x_0^t = x_1^t = x_{n_{t-1}+1}^{t-1}$
    for $m = 1$ to $n_t$ do
        $y_m^t = (1 + \mu_t) x_m^t - \mu_t x_{m-1}^t$
        Send $y_m^t$ to every node in $\mathcal{T}$.
        Each node $i$ computes $g_i = \frac{1}{K} \sum_{k=1}^{K} \nabla f_i(y_m^t + \gamma X_k)$, where $X_k \sim \mathcal{N}(0, I)$
        $x_{m+1}^t = y_m^t - \frac{\eta_t}{n} \sum_i g_i$
    end for
end for
return $\theta_T = x_T$
```

noise is unimprovable by modifying the noise distribution. This is in accordance with recent theoretical results of Bubeck et al. (2019) that imply that the $d^{1/4}$ multiplicative factor in Equation 4 is unimprovable in the regime $T \leq \sqrt{d}$ where T is the number of iterations of the algorithm.

Remark 4 *Due to its random nature, Alg. 1 is not* per se *a black-box procedure, and Theorems 8 and 9 of Section 3.2 do not apply to it. Lower bounds for random algorithms are more challenging and left for future work.*

Remark 5 *Only the convexity of the* global *function $\bar{f}$ is necessary for Alg. 1 to converge. For example, the local functions f_i may be non convex differentiable functions whose average is convex.*

3.1.2. Strongly Convex Case

The same smoothing strategy is also applicable for strongly convex objectives, and yields similar improvements in the convergence rate. Unfortunately, the analysis of randomized smoothing for strongly convex objectives is yet missing in the literature, and we thus need to use another convergence result. Alg. 2 applies Nesterov's accelerated gradient descent to the smoothed objective function $\bar{f}^{\gamma}$, leading to fast communication rates for strongly convex objectives. More specifically, we use the recent algorithm M-ASG by Serhat Aybat et al. (2019) that provides tight convergence rates for accelerated gradient descent under additive noise on the gradient.

Theorem 6 *Under global Lipschitz-continuity and strong convexity and for $\varepsilon < \bar{f}(\theta_0) - \bar{f}(\theta^*)$, Alg. 2 achieves an approximation error $\mathbb{E}\left[\bar{f}(\theta_T)\right] - \bar{f}(\theta^*)$ of at most ε in a time T_ε upper-bounded by*

$$O\left(\frac{L_g^2}{\alpha_g\varepsilon} + \sqrt{\frac{L_g^2}{\alpha_g\varepsilon}}(1+\Delta\tau)d^{1/4}\log\left(\frac{1}{\varepsilon}\right)\right).$$

More specifically, Alg. 2 completes its T iterations by time

$$T_\varepsilon \quad \leq \quad 2\left\lceil\frac{\sqrt{2}L_g c_{d,\varepsilon}}{\sqrt{\alpha_g\varepsilon}}\right\rceil\left\lceil\frac{70L_g c_{d,\varepsilon}^{-1}}{\sqrt{\alpha_g\varepsilon}}\right\rceil + 4\left\lceil\frac{\sqrt{2}L_g c_{d,\varepsilon}}{\sqrt{\alpha_g\varepsilon}}\right\rceil\Delta\tau\,, \tag{6}$$

where $c_{d,\varepsilon} = d^{1/4}\log\left(\frac{8\left(\bar{f}(\theta_0)-\bar{f}(\theta^*)\right)}{\varepsilon}\right)$. The proof is available in Appendix A.

While Alg. 2 relies on the two-level nested iterations of M-ASG, we believe that more simple algorithms may also lead to similar convergence rates, e.g. directly using Nesterov's gradient descent and its recent stochastic analysis by Vaswani et al. (2019). Note that Theorem 6, as well as all lower bounds for the strongly convex case, is an addition compared to our preliminary work (Scaman et al., 2018).

Remark 7 *The Lipschitz and convex setting of Section 3.1.1 can also be solved by Alg. 2 by adding a quadratic regularization $\frac{\alpha}{2}\|\theta\|_2^2$ with a sufficiently small α, although this would incur a multiplicative logarithmic factor compared to the convergence rate of Alg. 1.*

3.2. Optimal Convergence Rates

The following results provide oracle complexity lower bounds for centralized algorithms, and are proved in Appendix B. These lower bounds extend the communication complexity lower bounds for totally connected communication networks of Arjevani and Shamir (2015). Note that, in all following results, we consider that the dimension d is larger than both the number of communication and computation steps. This assumption is relatively standard in non-smooth optimization (Nesterov, 2004, Theorem 3.2.1) and valid for either small times or large dimensional problems. When the number of iterations is larger than the dimension d, linearly convergent algorithms exist even for convex non-smooth problems (Bubeck, 2015).

3.2.1. Non-Smooth Objective Functions

Theorem 8 *Let $\mathcal{G}$ be a network of computing units of size $n > 0$, and $L_g, R > 0$. There exists n convex functions $f_i : \mathbb{R}^d \to \mathbb{R}$ such that $\bar{f}$ is L_g-Lipschitz and, for any $t < \frac{d-2}{2}\min\{1, \Delta\tau\}$ and any black-box procedure one has, for all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta\in B_2(R)}\bar{f}(\theta) \geq \frac{RL_g}{36}\sqrt{\frac{1}{(1+\frac{t}{2\Delta\tau})^2} + \frac{1}{1+t}}\,. \tag{7}$$

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 2 + 2\max\{t, t/\Delta\tau\}$), Theorem 8 implies that, under global Lipschitz continuity, the time to reach a precision $\varepsilon > 0$ with any black-box procedure is lower bounded by

$$\Omega\bigg(\Big(\frac{RL_g}{\varepsilon}\Big)^2 + \frac{RL_g}{\varepsilon}\Delta\tau\bigg),$$

where the notation $g(\varepsilon) = \Omega(f(\varepsilon))$ stands for $\exists C > 0$ s.t. $\forall \varepsilon > 0, g(\varepsilon) \geq Cf(\varepsilon)$. This lower bound proves that the convergence rate of DRS in Equation 4 is optimal with respect to computation time and within a $d^{1/4}$ multiplicative factor of the optimal convergence rate with respect to communication.

Theorem 9 *Let $\mathcal{G}$ be a network of computing units of size $n > 0$, and $L_g, \alpha_g > 0$. There exists a convex set $\Theta \subset \mathbb{R}^d$ and n convex functions $f_i : \Theta \to \mathbb{R}$ such that $\bar{f}$ is L_g-Lipschitz and α_g-strongly convex and, for any $t < \frac{d-2}{2}\min\{1, \Delta\tau\}$ and any black-box procedure one has, for all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta\in\Theta} \bar{f}(\theta) \geq \frac{L_g^2}{108\alpha_g}\left(\frac{1}{(1+\frac{t}{2\Delta\tau})^2} + \frac{1}{1+t}\right). \tag{8}$$

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 2 + 2\max\{t, t/\Delta\tau\}$), Theorem 9 implies that, under global strong convexity and Lipschitz continuity, the time to reach a precision $\varepsilon > 0$ with any black-box procedure is lower bounded by

$$\Omega\bigg(\frac{L_g^2}{\alpha_g\varepsilon} + \sqrt{\frac{L_g^2}{\alpha_g\varepsilon}}\Delta\tau\bigg).$$

This lower bound proves that the convergence rate of the strongly convex version of DRS in Equation 6 is, up to logarithmic factors, optimal with respect to computation time and within a $d^{1/4}$ multiplicative factor of the optimal convergence rate with respect to communication.

The proof of Theorem 8 and Theorem 9 relies on the use of two objective functions: first, the standard worst case function used for single machine convex optimization (see, e.g., Bubeck, 2015) is used to obtain a lower bound on the local computation time of individual machines. Then, a second function first introduced by Arjevani and Shamir (2015) is split on the two most distant machines to obtain worst case communication times. By aggregating these two functions, a third one is obtained with the desired lower bound on the convergence rate. The complete proof is available in Appendix B.1. Finally, note that, due to its random nature, Alg. 1 is not *per se* a black-box procedure, and Theorems 8 and 9 do not apply to it. Lower bounds for random algorithms are more challenging and left for future work.

Remark 10 *The lower bounds also hold for the average of local parameters $\frac{1}{n}\sum_{i=1}^n \theta_i$, and more generally any parameter that can be computed using any vector of the local memories at time t: in Theorem 8, $\theta_{i,t}$ may be replaced by any θ_t such that*

$$\theta_t \in \mathrm{Span}\bigg(\bigcup_{i\in\mathcal{V}} \mathcal{M}_{i,t}\bigg).$$

3.2.2. Smooth Objective Functions

The following theorems provide lower bounds for smooth centralized optimization. The first is a novel result of this work, while the second was first derived by Scaman et al. (2017, Theorem 1) and is included in this section for the sake of completeness.

Theorem 11 *Let $\mathcal{G}$ be a network of computing units of size $n > 0$, and $\beta_g > 0$. There exists n convex functions $f_i : \mathbb{R}^d \to \mathbb{R}$ such that $\bar{f}$ is β_g-smooth and, for any $t \leq 1 + \frac{d-3}{2}(1 + \Delta\tau)$ and any black-box procedure one has, for all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta \in \mathbb{R}^d} \bar{f}(\theta) \geq \frac{3\beta_g \|\theta_{i,0} - \theta^*\|^2}{32\left(\frac{t-1}{1+\Delta\tau} + 1\right)^2}, \tag{9}$$

where θ^* is a minimizer of $\bar{f}$.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 3 + 2(t-1)/(1+\Delta\tau)$), Theorem 11 implies that, under global smoothness, the time to reach a precision $\varepsilon > 0$ with any black-box procedure is lower bounded by

$$\Omega\left(\sqrt{\frac{\beta_g \|\theta_{i,0} - \theta^*\|^2}{\varepsilon}}\Big(1 + \Delta\tau\Big)\right).$$

In order to simplify the proofs of the following theorem, and following the approach of Bubeck (2015), we will consider the limiting situation $d \to +\infty$. More specifically, we now assume that we are working in $\ell_2 = \{\theta = (\theta_k)_{k \in \mathbb{N}} \ : \ \sum_k \theta_k^2 < +\infty\}$ rather than $\mathbb{R}^d$.

Theorem 12 *Let $\mathcal{G}$ be a network of computing units of size $n > 0$, and $\beta_g \geq \alpha_g > 0$. There exists n convex functions $f_i : \ell_2 \to \mathbb{R}$ such that $\bar{f}$ is α_g strongly convex and β_g smooth, and for any $t \geq 0$ and any black-box procedure one has, for all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \bar{f}(\theta^*) \geq \frac{\alpha_g}{2}\left(1 - \frac{4}{\sqrt{\kappa_g}}\right)^{1 + \frac{t}{1+\Delta\tau}} \|\theta_{i,0} - \theta^*\|^2, \tag{10}$$

where $\kappa_g = \beta_g / \alpha_g$.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d \gg \max\{t, t/\Delta\tau\}$), Theorem 12 implies that, under global strong convexity and smoothness, the time to reach a precision $\varepsilon > 0$ with any black-box procedure is lower bounded by

$$\Omega\left(\sqrt{\kappa_g} \ln\left(\frac{1}{\varepsilon}\right)\Big(1 + \Delta\tau\Big)\right).$$

These two lower bounds prove that the naïve distribution of Nesterov's gradient descent is optimal for both smooth and smooth and strongly convex objective functions. Their proofs rely on splitting the functions used by Nesterov to prove oracle complexities for smooth optimization (Nesterov, 2004; Bubeck, 2015) on two nodes at distance Δ. One can show that most dimensions of the parameters $\theta_{i,t}$ will remain zero, and local gradient computations may only increase the number of non-zero dimensions by one. Finally, at least Δ communication rounds are necessary in-between every gradient computation, in order to share information between the two nodes. The complete proofs are available in Appendix B.2.

4. Decentralized Optimization under Local Regularity

In many practical scenarios, the network may be unknown or changing through time, and a local communication scheme is preferable to the *master/slave* approach of Alg. 1. Decentralized algorithms tackle this problem by replacing targeted communication by *local averaging* of the values of neighboring nodes (Boyd et al., 2006). More specifically, we now consider that, during a communication step, each machine i broadcasts a vector $x_i \in \mathbb{R}^d$ to its neighbors, then performs a weighted average of the values received from its neighbors:

$$\text{node } i \text{ sends } x_i \text{ to his neighbors and receives } \textstyle\sum_j W_{ji} x_j \,.$$

In order to ensure the efficiency of this communication scheme, we impose standard assumptions on the matrix $W \in \mathbb{R}^{n \times n}$, called the *gossip* matrix (Boyd et al., 2006; Scaman et al., 2017):

1. W is symmetric and positive semi-definite.

2. The kernel of W is the set of constant vectors: $\mathrm{Ker}(W) = \mathrm{Span}(\mathbb{1})$, where $\mathbb{1} = (1, ..., 1)^\top$.

3. W is defined on the edges of the network: $W_{ij} \neq 0$ only if $i = j$ or $(i, j) \in \mathcal{E}$.

This procedure, known as gossip algorithm (Boyd et al., 2006), is a standard method for averaging values across a network when its connectivity may vary through time. This approach was shown to be robust against machine failures, non-uniform latencies and asynchronous or time-varying graphs, and a large body of literature extended this algorithm to distributed optimization (Nedic and Ozdaglar, 2009; Duchi et al., 2012a; Wei and Ozdaglar, 2012; Shi et al., 2015; Jakovetić et al., 2015; Nedic et al., 2017; Aryan Mokhtari and Ribeiro, 2016). Note that these assumptions are implied by symmetry, stochasticity and positive eigengap on $I - W$. The convergence analysis of decentralized algorithms usually relies on the spectrum of the *gossip matrix* W used for communicating values in the network, and more specifically on the ratio between the second smallest and the largest eigenvalue of W, denoted $\gamma(W) = \lambda_{n-1}(W)/\lambda_1(W)$ and sometimes called *normalized eigengap*. More precisely, our lower bounds on the optimal convergence rate are obtained by replacing the diameter of the network with the quantity $\widetilde{\Delta} = 1/\sqrt{\gamma(W)}$, that we refer to as the *mixing time* due to its relation to the mixing time of random walks when W is the Laplacian matrix of the communication network.

To ease the presentation of this section, and contrary to Section 3, we first present the lower bounds for decentralized optimization in Section 4.1, before describing the corresponding optimal algorithms in Section 4.2.

Remark 13 *When communication is asymmetric, or the assumptions on the gossip matrix do not hold, traditional gossip methods fail and more advanced communication methods are required. We refer the reader to Nedic and Olshevsky (2015); Xi and Khan (2017); Nedic and Olshevsky (2016); Nedic et al. (2016) for asymmetric decentralized optimization algorithms.*

4.1. Optimal Convergence Rates

The following results provide oracle complexity lower bounds for decentralized algorithms, and are proved in Appendix C. Similarly to Section 3.2, we consider that the dimension d is larger than both the number of communication and computation steps. This assumption is relatively standard in non-smooth optimization (Nesterov, 2004, Theorem 3.2.1) and valid for either small times or large dimensional problems. When the number of iterations is larger than the dimension d, linearly convergent algorithms exist even for convex non-smooth problems (Bubeck, 2015).

4.1.1. Non-Smooth Objective Functions

Theorem 14 *Let $L_\ell, R > 0$ and $\widetilde{\Delta} \geq 1$. There exists a matrix W of mixing time $\widetilde{\Delta}$, and n convex and L_i-Lipschitz functions f_i, where n is the size of W, such that for all $t < \frac{d-2}{2}\min(1, \widetilde{\Delta}\tau)$ and all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta \in B_2(R)} \bar{f}(\theta) \geq \frac{RL_\ell}{108}\sqrt{\frac{1}{\left(1 + \frac{2t}{\widetilde{\Delta}\tau}\right)^2} + \frac{1}{1+t}}\,,$$

where $L_\ell = \sqrt{\frac{1}{n}\sum_i L_i^2}$.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 2 + 2\max\{t, t/\widetilde{\Delta}\tau\}$), Theorem 14 implies that, under local Lipschitz continuity and for a gossip matrix W with mixing time $\widetilde{\Delta}$, the time to reach a precision $\varepsilon > 0$ with any *decentralized* black-box procedure is lower-bounded by

$$\Omega\left(\left(\frac{RL_\ell}{\varepsilon}\right)^2 + \frac{RL_\ell}{\varepsilon}\widetilde{\Delta}\tau\right).$$

Theorem 15 *Let $L_\ell, \alpha_\ell > 0$ and $\widetilde{\Delta} \geq 1$. There exists a matrix W of size n and mixing time $\widetilde{\Delta}$, a convex set $\Theta \subset \mathbb{R}^d$ and n α_i-strongly convex and L_i-Lipschitz functions $f_i : \Theta \to \mathbb{R}$, such that for all $t < \frac{d-2}{2}\min(1, \widetilde{\Delta}\tau)$ and all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta \in \Theta} \bar{f}(\theta) \geq \frac{L_\ell^2}{648\alpha_\ell}\sqrt{\frac{1}{\left(1 + \frac{2t}{\widetilde{\Delta}\tau}\right)^2} + \frac{1}{1+t}}\,,$$

where $\alpha_\ell = \min_i \alpha_i$ and $L_\ell = \sqrt{\frac{1}{n}\sum_i L_i^2}$.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 2+2\max\{t, t/\widetilde{\Delta}\tau\}$), Theorem 15 implies that, under local strong convexity and Lipschitz continuity, and for a gossip matrix W with mixing time $\widetilde{\Delta}$, the time to reach a precision $\varepsilon > 0$ with any *decentralized* black-box procedure is lower-bounded by

$$\Omega\left(\frac{L_\ell^2}{\alpha_\ell\varepsilon} + \sqrt{\frac{L_\ell^2}{\alpha_\ell\varepsilon}}\widetilde{\Delta}\tau\right).$$

The proof of Theorem 14 relies on linear graphs (whose diameter is proportional to $\widetilde{\Delta}$ when the Laplacian matrix is used as gossip matrix) and Theorem 8. More specifically, a technical aspect of the proof consists in splitting the functions used in Theorem 8 on multiple nodes to obtain a dependency in L_ℓ instead of L_g. The complete derivation is available in Appendix C.1.

4.1.2. Smooth Objective Functions

The following theorems provide lower bounds for smooth centralized optimization. The first is a novel result of this work, while the second was first derived by Scaman et al. (2017, Theorem 2) and is included in this section for the sake of completeness.

Theorem 16 *Let $\beta_\ell > 0$ and $\widetilde{\Delta} \geq 1$. There exists a matrix W of mixing time $\widetilde{\Delta}$, and n convex and β_i-smooth functions f_i, where n is the size of W, such that for all $t \leq 1 + \frac{d-3}{2}(1 + \widetilde{\Delta}\tau)$ and all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \min_{\theta \in \mathbb{R}^d} \bar{f}(\theta) \geq \frac{3\beta_\ell \|\theta_{i,0} - \theta^*\|^2}{64\left(\frac{3(t-1)}{3+\widetilde{\Delta}\tau} + 1\right)^2},$$

where θ^* is a minimizer of $\bar{f}$ and $\beta_\ell = \max_i \beta_i$.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d > 3 + 2(t-1)/(1 + \widetilde{\Delta}\tau)$), Theorem 16 implies that, under local smoothness, and for a gossip matrix W with mixing time $\widetilde{\Delta}$, the time to reach a precision $\varepsilon > 0$ with any *decentralized* black-box procedure is lower-bounded by

$$\Omega\left(\sqrt{\frac{\beta_\ell \|\theta_{i,0} - \theta^*\|^2}{\varepsilon}}\left(1 + \widetilde{\Delta}\tau\right)\right).$$

Theorem 17 *Let $\alpha, \beta > 0$ and $\widetilde{\Delta} \geq 1$. There exists a gossip matrix W of mixing time $\widetilde{\Delta}$, and α-strongly convex and β-smooth functions $f_i : \ell_2 \to \mathbb{R}$ such that, for any $t \geq 0$ and any black-box procedure using W one has, for all $i \in [\![1, n]\!]$,*

$$\bar{f}(\theta_{i,t}) - \bar{f}(\theta^*) \geq \frac{\alpha}{2}\left(1 - \frac{16}{\sqrt{\kappa_\ell}}\right)^{1 + \frac{3t}{3 + \widetilde{\Delta}\tau}} \|\theta_{i,0} - \theta^*\|^2,$$

where $\kappa_\ell = \beta/\alpha$ is the local condition number.

Assuming that the dimension d is large compared to the characteristic values of the problem (i.e., $d \gg \max\{t, t/\widetilde{\Delta}\tau\}$), Theorem 17 implies that, under local strong convexity and smoothness, and for a gossip matrix W with mixing time $\widetilde{\Delta}$, the time to reach a precision $\varepsilon > 0$ with any *decentralized* black-box procedure is lower-bounded by

$$\Omega\left(\sqrt{\kappa_l}\ln\left(\frac{1}{\varepsilon}\right)\left(1 + \widetilde{\Delta}\tau\right)\right).$$

The proofs of Theorem 16 and Theorem 17 rely on the same technique as that of Theorem 14 and Theorem 15. The complete proof is available in Appendix C.2.

We will see in the next section that this lower bound is met for a novel decentralized algorithm called *multi-step primal-dual* (MSPD) and based on the dual formulation of the optimization problem. Note that these results provide optimal convergence rates with respect to κ_l and $\widetilde{\Delta}$, but do not imply that $\widetilde{\Delta}$ is the right quantity to consider on general graphs. The quantity $\widetilde{\Delta}$ may indeed be very large compared to Δ, for example for star networks, for which $\Delta = 2$ and $\widetilde{\Delta} = \sqrt{n}$. However, on many simple networks, the diameter Δ and the eigengap of the Laplacian matrix are tightly connected, and $\Delta \approx \widetilde{\Delta}$. For example, for linear graphs, $\Delta = n - 1$ and $\widetilde{\Delta} \approx 2n/\pi$, for totally connected networks, $\Delta = 1$ and $\widetilde{\Delta} = 1$, and for regular networks, $\widetilde{\Delta} \geq \frac{\Delta}{2\sqrt{2}\ln_2 n}$ (Alon and Milman, 1985). Finally, note that the case of totally connected networks corresponds to a previous complexity lower bound on communications proven by Arjevani and Shamir (2015), and is equivalent to our result for centralized algorithms with $\Delta = 1$.

4.2. Optimal Decentralized Algorithms

In this section, we present optimal algorithms for the decentralized setting.

4.2.1. Non-Smooth Objective Functions

We now provide an optimal decentralized optimization algorithm under (A2). This algorithm is closely related to the primal-dual algorithm proposed by Lan et al. (2017), which we modify by the use of accelerated gossip using Chebyshev polynomials as done by Scaman et al. (2017).

First, following Jakovetić et al. (2015), we formulate our optimization problem in Equation 1 as the saddle-point problem in Equation 11 below, by considering the equivalent problem of minimizing $\frac{1}{n}\sum_{i=1}^n f_i(\theta_i)$ over $\Theta = (\theta_1, \dots, \theta_n) \in \mathcal{K}^n$ with the constraint that $\theta_1 = \dots = \theta_n$, or equivalently $\Theta A = 0$, where A is a square root of the symmetric matrix W. Through Lagrangian duality, we therefore get the equivalent problem:

$$\min_{\Theta \in \mathcal{K}^n} \max_{\Lambda \in \mathbb{R}^{d \times n}} \frac{1}{n} \sum_{i=1}^n f_i(\theta_i) - \operatorname{tr} \Lambda^\top \Theta A \,. \tag{11}$$

We solve it by applying the algorithm of Chambolle and Pock (2011), which is both simple and well tailored to our problem—we could alternatively apply composite Mirror-Prox (He et al., 2015): (a) it is an accelerated method for saddle-point problems, (b) it allows for composite problems with a sum of non-smooth and smooth terms, (c) it provides a primal-dual gap that can easily be extended to the case of approximate proximal operators. At each iteration t, with initialization $\Lambda^0 = 0$ and $\Theta^0 = \Theta^{-1} = (\theta_0, \dots, \theta_0)$:

$$\begin{aligned}
(a) \quad \Lambda^{t+1} &= \Lambda^t - \sigma(2\Theta^t - \Theta^{t-1})A \\
(b) \quad \Theta^{t+1} &= \operatorname*{argmin}_{\Theta \in \mathcal{K}^n} \frac{1}{n} \sum_{i=1}^n f_i(\theta_i) - \operatorname{tr} \Theta^\top \Lambda^{t+1} A^\top + \frac{1}{2\eta} \operatorname{tr}(\Theta - \Theta^t)^\top (\Theta - \Theta^t)\,,
\end{aligned}$$

where the gain parameters η, σ are required to satisfy $\sigma\eta\lambda_1(W) \leq 1$. We implement the algorithm with the variables Θ^t and $Y^t = \Lambda^t A^\top = (y_1^t, \dots, y_n^t) \in \mathbb{R}^{d \times n}$, for which all updates

Algorithm 3 multi-step primal-dual algorithm

```
Input: approximation error $\varepsilon > 0$, gossip matrix $W \in \mathbb{R}^{n \times n}$,
    $T = \lceil \frac{2RL_\ell \widetilde{\Delta}}{\varepsilon} \rceil$, $M = \lceil \frac{2RL_\ell}{\varepsilon \widetilde{\Delta}} \rceil$, $\eta = \frac{nR}{L_\ell \widetilde{\Delta}}$, $\sigma = \frac{1}{\eta \lambda_1(W)}$ .
Output: optimizer $\bar{\theta}_T$
$\Theta_0 = 0$, $\Theta_{-1} = 0$, $Y_0 = 0$
for $t = 0$ to $T - 1$ do
    $Y^{t+1} = Y^t - \sigma(2\Theta^t - \Theta^{t-1})W$
    $\tilde{\Theta}^0 = \Theta^t$
    for $m = 0$ to $M - 1$ do
        $\tilde{\theta}_i^{m+1} = \frac{m}{m+2} \tilde{\theta}_i^m - \frac{2}{m+2} [\frac{\eta}{n} \nabla f_i(\tilde{\theta}_i^m) - \eta y_i^{t+1} - \theta_i^t]$, $\forall i \in [\![1, n]\!]$
    end for
    $\Theta^{t+1} = \tilde{\Theta}^M$
end for
return $\bar{\theta}_T = \frac{1}{T} \frac{1}{n} \sum_{t=1}^{T} \sum_{i=1}^{n} \theta_i^t$
```

can be made locally: since $AA^\top = W$, they now become

$$\begin{array}{lrcl} (a') & Y^{t+1} & = & Y^t - \sigma(2\Theta^t - \Theta^{t-1})W \\ (b') & \theta_i^{t+1} & = & \operatorname*{argmin}_{\theta_i \in \mathcal{K}} \dfrac{1}{n} f_i(\theta_i) - \theta_i^\top y_i^{t+1} + \dfrac{1}{2\eta} \|\theta_i - \theta_i^t\|^2, \forall i \in [\![1, n]\!] \, . \end{array} \tag{12}$$

The step (b') still requires a proximal step for each function f_i. We approximate it by the outcome of the subgradient method run for M steps, with a step-size proportional to $2/(m+2)$ as suggested by Lacoste-Julien et al. (2012). That is, initialized with $\tilde{\theta}_i^0 = \theta_i^t$, it performs the iterations

$$\tilde{\theta}_i^{m+1} = \frac{m}{m+2} \tilde{\theta}_i^m - \frac{2}{m+2} \left[\frac{\eta}{n} \nabla f_i(\tilde{\theta}_i^m) - \eta y_i^{t+1} - \theta_i^t \right], \quad m = 0, \ldots, M-1. \tag{13}$$

We thus replace the step (b') by running M steps of the subgradient method to obtain $\tilde{\theta}_i^M$.

Theorem 18 *Under local Lipschitz continuity, the approximation error with the iterative algorithm of Equation 12 and 13 after T iterations and using M subgradient steps per iteration is bounded by*

$$\bar{f}(\bar{\theta}_T) - \min_{\theta \in \mathcal{K}} \bar{f}(\theta) \leq RL_\ell \Big(\frac{\widetilde{\Delta}}{T} + \frac{1}{M \widetilde{\Delta}} \Big).$$

Theorem 18 implies that the proposed algorithm achieves an error of at most ε in a time no larger than

$$O\bigg(\Big(\frac{RL_\ell}{\varepsilon} \Big)^2 + \frac{RL_\ell}{\varepsilon} \widetilde{\Delta} \tau \bigg), \tag{14}$$

which is optimal due to Theorem 14. This algorithm, called *multi-step primal-dual* (MSPD), is described in Alg. 3.

Remark 19 *It is clear from the algorithm's description that it completes its T iterations by time*

$$T_\varepsilon \quad \leq \quad \left\lceil\frac{2RL_\ell\widetilde{\Delta}}{\varepsilon}\right\rceil\left\lceil\frac{2RL_\ell}{\varepsilon\widetilde{\Delta}}\right\rceil + \left\lceil\frac{2RL_\ell\widetilde{\Delta}}{\varepsilon}\right\rceil\tau\,. \tag{15}$$

To obtain the average of local parameters $\bar{\theta}_T = \frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \theta_i$, one can then rely on accelerated gossip (Auzinger, 2011; Arioli and Scott, 2014) to average over the network the individual nodes' time averages. This leads to a time $O\big(\ln\big(\frac{RL_\ell}{\varepsilon}\big)\widetilde{\Delta}\tau\big)$ to ensure that each node reaches a precision ε on the objective function (see Boyd et al., 2006, for more details on the linear convergence of gossip), which is negligible compared to Equation 14.

Remark 20 *A stochastic version of the algorithm is also possible by considering stochastic oracles on each f_i and using stochastic subgradient descent instead of the subgradient method.*

Remark 21 *In the more general context where node compute times ρ_i are not necessarily all equal to 1, we may still apply Alg. 3, where now the number of subgradient iterations performed by node i is M/ρ_i rather than M. The proof of Theorem 18 also applies, and now yields the modified upper bound on time to reach precision ε:*

$$O\left(\left(\frac{RL_c}{\varepsilon}\right)^2 + \frac{RL_\ell}{\varepsilon}\widetilde{\Delta}\tau\right), \tag{16}$$

where $L_c^2 = \frac{1}{n}\sum_{i=1}^n \rho_i L_i^2$.

4.2.2. Other Regularity Assumptions

While this primal-dual approach could in theory be applied to all remaining three settings (strongly convex, smooth, and both strongly convex and smooth), the convergence rates of the Chambolle-Pock algorithm are weaker in such settings, and lead to convergence rates with respect to the distance to the optimum in squared norm $\|\theta_t - \theta^*\|^2$ (see Chambolle and Pock, 2011, Theorems 2 and 3). These issues could be avoided using the stronger results of composite Mirror-Prox (He et al., 2015), but would require substantial modifications to the algorithm. Fortunately, these cases were recently investigated by Dvinskikh and Gasnikov (2019), who designed optimal algorithms up to logarithmic factors for each decentralized communication setting (*R-Sliding* for strongly convex objectives and *PSTM* for smooth objectives). We thus refer to their work for the remaining cases.

5. Conclusion

In this paper, we provide optimal convergence rates for convex distributed optimization for two communication schemes (centralized and decentralized) and under four regularity assumptions on the objective function: Lipschitz continuity, strong convexity, smoothness, and both strong convexity and smoothness. Under decentralized communication, we provide optimal convergence rates that depend on the *local* regularity characteristics and *mixing time* of the gossip matrix. Moreover, we also provide the first optimal decentralized algorithm, called *multi-step primal-dual* (MSPD).

Under centralized communication, we first show that the naïve distribution of accelerated gradient descent is optimal for smooth functions. Then, we provide lower complexity bounds for non-smooth functions that depends on the (global) characteristics of the objective function, as well as a distributed version of the smoothing approach of Duchi et al. (2012b) and show that this algorithm is within a $d^{1/4}$ multiplicative factor of the optimal convergence rate.

In both settings, the optimal convergence rate for non-smooth distributed optimization exhibits two different speeds: a slow rate in $\Theta(1/\sqrt{t})$ with respect to local computations and a fast rate in $\Theta(1/t)$ due to communication. Intuitively, communication is the limiting factor in the initial phase of optimization. However, its impact decreases with time and, for the final phase of optimization, local computation time is the main limiting factor.

The analysis presented in this paper allows several natural extensions, including time-varying communication networks, asynchronous algorithms, in the line of Hendrikx et al. (2019b), stochastic settings (see, e.g., Hendrikx et al., 2019a; Stich, 2018), and an analysis of unequal node compute speeds going beyond Remark 21. Moreover, despite the efficiency of DRS, finding an optimal algorithm under the global regularity assumption remains an open problem and would make a notable addition to this work.

Acknowledgments

We acknowledge support from the European Research Council (grant SEQUOIA 724063).

Appendix A. Proof of the Convergence Rate of DRS (Theorems 2 and 6)

In this appendix, we provide detailed proofs of Theorems 2 and 6, that is, for the convex and strongly-convex cases.

A.1. Convex Case

Corollary 2.4 of Duchi et al. (2012b) gives, with the appropriate choice of gradient step η_t and smoothing γ_t,

$$\mathbb{E}\left[\bar{f}(\theta_T)\right] - \min_{\theta\in\mathcal{K}} \bar{f}(\theta) \leq \frac{10RL_g d^{1/4}}{T} + \frac{5RL_g}{\sqrt{TK}}.$$

Thus, to reach a precision $\varepsilon > 0$, we may set $T = \left\lceil \frac{20RL_g d^{1/4}}{\varepsilon} \right\rceil$ and $K = \left\lceil \frac{5RL_g d^{-1/4}}{\varepsilon} \right\rceil$, leading to the desired bound on the time $T_\varepsilon = T(2\Delta\tau + K)$ to reach ε.

A.2. Strongly Convex Case

Using Corollary 3.9 of Serhat Aybat et al. (2019) to optimize $\bar{f}^\gamma$, the number of iterations n_ε to reach a precision $\varepsilon/2$ is bounded by

$$n_\varepsilon \leq \left\lceil \sqrt{\frac{2L_g^2}{\alpha_g \varepsilon}} c_{d,\varepsilon} \right\rceil + 32(1 + \log(8)) \frac{L_g^2}{\alpha_g K \varepsilon} \,,$$

where $c_{d,\varepsilon} = d^{1/4}\log\left(\frac{8\left(\bar{f}(\theta_0)-\bar{f}(\theta^*)\right)}{\varepsilon}\right)$. Since $\bar{f}^\gamma$ is an $L_g\gamma\sqrt{d}$-approximation of $\bar{f}$ (see Lemma 1), setting $\gamma = \frac{\varepsilon}{2L_g\sqrt{d}}$ leads to a precision ε on $\bar{f}$ after n_ε iterations. Finally, each iteration takes a time $n_\varepsilon(K + 2\Delta\tau)$, and setting $K = \left\lceil \frac{70 L_g c_{d,\varepsilon}^{-1}}{\sqrt{\alpha_g \varepsilon}} \right\rceil$ leads to the desired result.

Appendix B. Proof of the Centralized Lower Bounds (Theorems 8, 9, 11 and 12)

The proofs of lower bounds all rely on the fact that, for well-chosen local functions, most of the coordinates of the vectors in the memory of any node will remain equal to 0.

Definition 22 *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph of n nodes, $I_0, I_1 \subset \mathcal{V}$ and $\|x\|_{nz} = \max\{k \in \mathbb{N} \;:\; x_k \neq 0\}$ the largest non-zero coordinate of x. Local functions $(f_i)_{i\in[\![1,n]\!]}$ are called* (I_0, I_1)-zero preserving *if, for all $x \in \mathbb{R}^d$ and $i \in [\![1,n]\!]$,*

$$\|\nabla f_i(x)\|_{nz} \leq \begin{cases} \|x\|_{nz} + \mathbb{1}\{\|x\|_{nz} \equiv 0 \ mod\ 2\} & if\ i \in I_0, \\ \|x\|_{nz} + \mathbb{1}\{\|x\|_{nz} \equiv 1 \ mod\ 2\} & if\ i \in I_1, \\ \|x\|_{nz} & otherwise. \end{cases}$$

In other words, local gradients can only increase even dimensions for nodes in I_0 and odd dimensions for nodes in I_1. With such objective functions, information needs to travel from I_0 to I_1 at each iteration in order to increase the number of non-zero coordinates.

Lemma 23 *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph of n nodes, $I_0, I_1 \subset \mathcal{V}$ and $k_{i,t} = \max\{k \in \mathbb{N} \;:\; \exists\theta \in \mathcal{M}_{i,t}$ s.t. $\theta_k \neq 0\}$ be the last non-zero coordinate of a vector in the memory of node i at time t. If the local functions are (I_0, I_1)-zero preserving, then*

$$\forall i \in \mathcal{V}, k_{i,t} \leq \left\lfloor \frac{t-1}{1 + d(I_0, I_1)\tau} \right\rfloor + 1\,, \tag{17}$$

where $d(I_0, I_1)$ is the shortest-path distance between I_0 and I_1 in $\mathcal{G}$.

Proof Since the local functions are zero preserving, we have, under any black-box procedure and for any local computation step,

$$k_{i,t+1} \leq \begin{cases} k_{i,t} + \mathbb{1}\{k_{i,t} \equiv 0 \text{ mod } 2\} & \text{if } i \in I_0, \\ k_{i,t} + \mathbb{1}\{k_{i,t} \equiv 1 \text{ mod } 2\} & \text{if } i \in I_1, \\ k_{i,t} & \text{otherwise.} \end{cases}$$

Thus, in order to reach the third coordinate, algorithms must first perform one local computation in I_0, then $d(I_0, I_1)$ communication steps in order for a node in I_1 to have a non-zero second coordinate, and finally, one local computation in I_1. Accordingly, one must perform at least k local computation steps and $(k-1)d(I_0, I_1)$ communication steps to achieve $k_{i,t} \geq k$ for at least one node $i \in \mathcal{V}$, and thus, for any $k \in \mathbb{N}^*$,

$$\forall t < 1 + (k-1)(1 + d(I_0, I_1)\tau),\ k_{i,t} \leq k - 1\,,$$

which leads to the desired result. ■

B.1. Non-Smooth Lower Bounds

Let $i_0 \in \mathcal{V}$ and $i_1 \in \mathcal{V}$ be two nodes at distance Δ. Following Nesterov (2004), the function used by Bubeck (2015) to prove the oracle complexity for non-smooth convex functions is

$$g_1(\theta) = \delta \max_{i \in [\![1,t]\!]} \theta_i + \frac{\alpha}{2}\|\theta\|_2^2.$$

By considering this function on a single node (e.g. i_0), at least $O\left(\left(\frac{RL}{\varepsilon}\right)^2\right)$ subgradients will be necessary to obtain a precision ε. Moreover, we also split the difficult function used by Arjevani and Shamir (2015)

$$g_2(\theta) = \gamma \sum_{i=1}^{t} |\theta_{i+1} - \theta_i| - \beta\theta_1 + \frac{\alpha}{2}\|\theta\|_2^2,$$

on the two extremal nodes i_0 and i_1 in order to ensure that communication is necessary between the most distant nodes of the network. The final function that we consider is, for all $j \in [\![1, n]\!]$,

$$f_j(\theta) = \begin{cases} \frac{\alpha}{2}\|\theta\|_2^2 + n\left[\gamma \sum_{i=1}^{k} |\theta_{2i+1} - \theta_{2i}| - \beta\theta_1\right] & \text{if } j = i_0, \\ \frac{\alpha}{2}\|\theta\|_2^2 + n\left[\gamma \sum_{i=1}^{k} |\theta_{2i} - \theta_{2i-1}| + \delta \max_{i \in [\![2k+2, 2k+1+l]\!]} \theta_i\right] & \text{if } j = i_1, \\ \frac{\alpha}{2}\|\theta\|_2^2 & \text{otherwise,} \end{cases} \tag{18}$$

where $\gamma, \delta, \beta, \alpha > 0$ and $k, l \geq 0$ are parameters of the function satisfying $2k + l < d$. The objective function is thus

$$\bar{f}(\theta) = \gamma \sum_{i=1}^{2k} |\theta_{i+1} - \theta_i| - \beta\theta_1 + \delta \max_{i \in [\![2k+2, 2k+1+l]\!]} \theta_i + \frac{\alpha}{2}\|\theta\|_2^2.$$

First, note that reordering the coordinates of θ between θ_2 and θ_{2k+1} in a decreasing order can only decrease the value function $\bar{f}(\theta)$. Hence, the optimal value θ^* verifies this constraint and

$$\bar{f}(\theta^*) = -\gamma\theta^*_{2k+1} - (\beta - \gamma)\theta^*_1 + \delta \max_{i \in [\![2k+2, 2k+1+l]\!]} \theta^*_i + \frac{\alpha}{2}\|\theta^*\|_2^2.$$

Moreover, at the optimum, all the coordinates between θ_2 and θ_{2k+1} are equal, all the coordinates between θ_{2k+2} and θ_{2k+1+l} are also equal, and all the coordinates after θ_{2k+1+l} are zero. Hence

$$\bar{f}(\theta^*) = -\gamma\theta^*_{2k+1} - (\beta - \gamma)\theta^*_1 + \delta\theta^*_{2k+2} + \frac{\alpha}{2}\left({\theta^*_1}^2 + 2k{\theta^*_{2k+1}}^2 + l{\theta^*_{2k+2}}^2\right),$$

and optimizing over $\theta_1^* \geq \theta_{2k+1}^* \geq 0 \geq \theta_{2k+2}^*$ leads to, when $\beta \geq \gamma(1 + \frac{1}{2k})$,

$$\bar{f}(\theta^*) = \frac{-1}{2\alpha}\left[(\beta - \gamma)^2 + \frac{\gamma^2}{2k} + \frac{\delta^2}{l}\right].$$

Now note that the objective function is the sum of two functions, one function on $(\theta_1, ..., \theta_{2k+1})$ that is (i_0, i_1)-zero preserving, and another function on $(\theta_{2k+2}, ..., \theta_{2k+1+l})$ that is (i_1, i_1)-zero preserving. As a result, Lemma 23 implies that, if $l > \lfloor t-1 \rfloor + 1$, we have $\theta_{t,2k+1+l} = 0$ and thus

$$\max_{i \in [\![2k+2, 2k+1+l]\!]} \theta_{t,i} \geq 0 \ .$$

Moreover, if $2k+1 > \left\lfloor \frac{t-1}{1+\Delta\tau} \right\rfloor + 1$, then $\theta_{t,2k+1} = 0$, and thus

$$\begin{array}{rcl} \bar{f}(\theta_t) & \geq & \min_{\theta \in \mathbb{R}^d} -(\beta - \gamma)\theta_1 + \frac{\alpha}{2}\|\theta\|_2^2 \\ & \geq & \frac{-(\beta-\gamma)^2}{2\alpha} \ . \end{array}$$

Hence, we have,

$$\bar{f}(\theta_t) - \bar{f}(\theta^*) \geq \frac{1}{2\alpha}\left[\frac{\gamma^2}{2k} + \frac{\delta^2}{l}\right].$$

Optimizing $\bar{f}$ over a ball of radius $R \geq \|\theta^*\|_2$ thus leads to the previous approximation error bound, and we choose

$$R^2 = \|\theta^*\|_2^2 = \frac{1}{\alpha^2}\left[(\beta - \gamma)^2 + \frac{\gamma^2}{2k} + \frac{\delta^2}{l}\right].$$

Finally, the Lipschitz constant of the objective function $\bar{f}$ on a ball of radius R is

$$L_g = \beta + 2\sqrt{2k+1}\gamma + \delta + \alpha R,$$

and setting the parameters of $\bar{f}$ to $\beta = \gamma(1 + \frac{1}{\sqrt{2k}})$, $\delta = \frac{L_g}{9}$, $\gamma = \frac{L_g}{9\sqrt{k}}$, $l = \lfloor t \rfloor + 1$, and $k = \left\lfloor \frac{t}{2\Delta\tau} \right\rfloor + 1$ leads to $l > \lfloor t-1 \rfloor + 1$ and $2k+1 > \left\lfloor \frac{t-1}{1+\Delta\tau} \right\rfloor + 1$, and

$$\bar{f}(\theta_t) - \bar{f}(\theta^*) \geq \frac{RL_g}{36}\sqrt{\frac{1}{(1 + \frac{t}{2\Delta\tau})^2} + \frac{1}{1+t}},$$

while $\bar{f}$ is L_g-Lipschitz and $\|\theta^*\|_2 \leq R$.

Moreover, $\bar{f}$ is α-strongly convex and rewriting the radius R is terms of $\alpha_g = \alpha$ and L_g gives

$$R = \frac{L_g}{3\alpha_g}\sqrt{\frac{1}{k^2} + \frac{1}{l}},$$

and thus

$$\bar{f}(\theta_t) - \bar{f}(\theta^*) \geq \frac{L_g^2}{108\alpha}\left(\frac{1}{(1 + \frac{t}{2\Delta\tau})^2} + \frac{1}{1+t}\right).$$

B.2. Smooth Lower Bounds

Both proofs of smooth lower bounds rely on splitting the function used by Nesterov to prove oracle complexities for smooth optimization (Nesterov, 2004; Bubeck, 2015).

B.2.1. Convex and Smooth Case

Let $\beta > 0$, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ a graph and $i_0 \in \mathcal{V}$ and $i_1 \in \mathcal{V}$ two nodes at distance Δ. Let, for all $j \in \mathcal{V}$, $f_j : \mathbb{R}^d \to \mathbb{R}$ be the functions defined as:

$$f_j(\theta) = \begin{cases} \frac{n\beta}{8}\left[\sum_{i=1}^k (\theta_{2i+1} - \theta_{2i})^2 + \theta_1^2 - 2\theta_1\right] & \text{if } j = i_0 \\ \frac{n\beta}{8}\left[\sum_{i=1}^k (\theta_{2i} - \theta_{2i-1})^2 + \theta_{2k+1}^2\right] & \text{if } j = i_1 \\ 0 & \text{otherwise} \end{cases}, \tag{19}$$

where $k \in \mathbb{N}$ is a parameter of the function. The objective function is thus

$$\bar{f}(\theta) = \frac{\beta}{8}\left[\sum_{i=1}^{2k} (\theta_{i+1} - \theta_i)^2 + \theta_1^2 + \theta_{2k+1}^2 - 2\theta_1\right].$$

First, note that $\nabla \bar{f}(\theta) = \frac{\beta}{8}(M\theta - 2e_1)$ where $M = \begin{pmatrix} M' & 0 \\ 0 & 0 \end{pmatrix}$ and $M' \in \mathbb{R}^{(2k+1)\times(2k+1)}$ is a tridiagonal matrix with 2 on the diagonal and -1 on the upper and lower diagonals. A simple calculation shows that $0 \preceq M \preceq 4I$, and thus $\bar{f}$ is β-smooth. The optimum of $\bar{f}$ is obtained for $\theta_i^* = 1 - \frac{i}{2k+2}$, and

$$\bar{f}(\theta^*) = -\frac{\beta}{8}\left(1 - \frac{1}{2k+2}\right).$$

Moreover, since the local functions are (i_0, i_1)-zero preserving, Lemma 23 implies that, for all $i > k_t = \left\lfloor \frac{t-1}{1+\Delta\tau} \right\rfloor + 1$, one has $\theta_{t,i} = 0$, and thus

$$\bar{f}(\theta_t) \geq -\frac{\beta}{8}\left(1 - \frac{1}{k_t + 1}\right).$$

Choosing $k = k_t$ directly implies

$$\bar{f}(\theta_t) - \bar{f}(\theta^*) \geq \frac{\beta}{16(k_t + 1)} \geq \frac{\beta}{16\left(\frac{t-1}{1+\Delta\tau} + 2\right)}.$$

Finally, noting that $\|\theta_0 - \theta^*\|^2 = \|\theta^*\|^2 \leq \frac{2(k+1)}{3}$ leads to the desired result.

B.2.2. Strongly Convex and Smooth Case

Let $\beta \geq \alpha > 0$, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ a graph and $i_0 \in \mathcal{V}$ and $i_1 \in \mathcal{V}$ two nodes at distance Δ. Let, for all $j \in \mathcal{V}$, $f_j : \ell_2 \to \mathbb{R}$ be the functions defined as:

$$f_j(\theta) = \begin{cases} \frac{\alpha}{2}\|\theta\|_2^2 + n\frac{\beta-\alpha}{8}(\theta^\top M_1 \theta - 2\theta_1) & \text{if } j = i_0, \\ \frac{\alpha}{2}\|\theta\|_2^2 + n\frac{\beta-\alpha}{8}\theta^\top M_2 \theta & \text{if } j = i_1, \\ \frac{\alpha}{2}\|\theta\|_2^2 & \text{otherwise,} \end{cases} \tag{20}$$

where $M_2 : \ell_2 \to \ell_2$ is the infinite block diagonal matrix with $\left(\begin{array}{cc} 1 & -1 \\ -1 & 1 \end{array}\right)$ on the diagonal, and $M_1 = \left(\begin{array}{cc} 1 & 0 \\ 0 & M_2 \end{array}\right)$. First, note that, since $0 \preceq M_1 + M_2 \preceq 4I$, $\bar{f} = \frac{1}{n}\sum_{i=1}^n f_i$ is α-strongly convex and β-smooth. Since the local functions are (i_0, i_1)-zero preserving, Lemma 23 implies that, for all $i > k_t = \left\lfloor \frac{t-1}{1+\Delta\tau} \right\rfloor + 1$, one has $\theta_{t,i} = 0$, and thus

$$\|\theta_t - \theta^*\|_2^2 \geq \sum_{i=k_t+1}^{+\infty} \theta_i^{*2}\,. \tag{21}$$

Since $\bar{f}$ is α-strongly convex, we also have

$$\bar{f}(\theta_t) - \bar{f}(\theta^*) \geq \frac{\alpha}{2}\|\theta_t - \theta^*\|_2^2. \tag{22}$$

Finally, the solution of the global problem $\min_\theta \bar{f}(\theta)$ is $\theta_i^* = \left(\frac{\sqrt{\beta}-\sqrt{\alpha}}{\sqrt{\beta}+\sqrt{\alpha}}\right)^i$. Combining this result with Equations 17, 21 and 22 leads to the desired inequality.

Appendix C. Proof of Decentralized Lower Bounds (Theorems 14, 15, 16 and 17)

Lower bounds for decentralized optimization are proved using the centralized setting on linear graphs and splitting the worst-case objective functions on multiple nodes. This allows to have both local characteristics approximately equal to global characteristics (i.e., $L_\ell \approx L_g$, $\alpha_\ell \approx \alpha_g$, $\beta_\ell \approx \beta_g$ and $\kappa_\ell \approx \kappa_g$) and the mixing time approximately equal to the diameter of the graph.

Lemma 24 *Let $\widetilde{\Delta} \geq 1$. There exists a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of size n and a gossip matrix $W \in \mathbb{R}^{n\times n}$ on this graph such that its mixing time is $\widetilde{\Delta}$. Moreover, there exists two subsets of nodes $I_0, I_1 \subset \mathcal{V}$ of size $|I_0| = |I_1| \geq n/4$ and such that*

$$d(I_0, I_1) \geq \frac{\sqrt{2}}{3}\widetilde{\Delta}\,, \tag{23}$$

where $d(I_0, I_1)$ is the shortest-path distance in $\mathcal{G}$ between the two sets.

Proof Let $\gamma = 1/\widetilde{\Delta}^2$ be the (desired) eigengap of the gossip matrix. First of all, when $\gamma \geq 1/3$, we consider the totally connected network of 3 nodes, reweight only the edge (v_1, v_3) by $a \in [0,1]$, and let W_a be its Laplacian matrix. If $a = 1$, then the network is totally connected and $\gamma(W_a) = 1$. If, on the contrary, $a = 0$, then the network is a linear graph and $\gamma(W_a) = 1/3$. Thus, by continuity of the eigenvalues of a matrix, there exists a value $a \in [0,1]$ such that $\gamma(W_a) = \gamma$ and Equation 23 is trivially verified. Otherwise, let $x_n = \frac{1-\cos(\frac{\pi}{n})}{1+\cos(\frac{\pi}{n})}$ be a decreasing sequence of positive numbers. Since $x_3 = 1/3$ and $\lim_n x_n = 0$, there exists $n \geq 3$ such that $x_n \geq \gamma > x_{n+1}$. Let $\mathcal{G}$ be the linear graph of size n ordered from node v_1 to v_n, and weighted with $w_{i,i+1} = 1 - a\mathbb{1}\{i = 1\}$. If we take W_a as the Laplacian of the weighted graph $\mathcal{G}$, a simple calculation gives that, if $a = 0$, $\gamma(W_a) = x_n$ and, if $a = 1$, the network is disconnected and $\gamma(W_a) = 0$. Thus, there exists a value $a \in [0,1]$ such that $\gamma(W_a) = \gamma$. By definition of n, one has

$$\gamma > x_{n+1} \geq \frac{2}{(n+1)^2}\,. \tag{24}$$

We now consider $I_0 = \llbracket 1, m \rrbracket$ and $I_1 = \llbracket n-m+1, n \rrbracket$ where $m = \lfloor \frac{n+1}{3} \rfloor$. When $\gamma < 1/3$, $\mathcal{G}$ is linear and the distance $d(I_0, I_1)$ between the two sets I_0 and I_1 is thus bounded by

$$d(I_0, I_1) = n - 2m + 1 \geq \frac{n+1}{3},$$

and using Equation 24 leads to the desired inequality. For $\gamma \geq 1/3$, $\mathcal{G}$ is a triangle and Equation 23 also trivially holds. Finally, since, for all $\gamma \in (0,1]$, $n \geq 3$, we have $m = \lfloor \frac{n+1}{3} \rfloor \geq \frac{n}{4}$. ■

Since all centralized lower bounds depend on splitting the objective function $\bar{f}(\theta) = \frac{f_1(\theta)+f_2(\theta)}{2}$ on two nodes at distance Δ, Lemma 24 allows to replace Δ by $\frac{\sqrt{2}}{3}\widetilde{\Delta}$. Finally, all lower bounds are obtained by splitting the functions f_1 and f_2 on all I_0 and I_1 nodes, respectively, in order to decrease local regularity characteristics.

C.1. Non-Smooth Lower Bounds

We now consider the local functions of Equation 18 split on I_0 and I_1 of Lemma 24:

$$f_j(\theta) = \begin{cases} \frac{\alpha}{2}\|\theta\|_2^2 + \frac{n}{m}\left[\gamma \sum_{i=1}^k |\theta_{2i} - \theta_{2i-1}| + \delta \max_{i \in \llbracket 2k+2, 2k+1+l \rrbracket} \theta_i\right] & \text{if } j \in I_0, \\ \frac{\alpha}{2}\|\theta\|_2^2 + \frac{n}{m}\left[\gamma \sum_{i=1}^k |\theta_{2i+1} - \theta_{2i}| - \beta\theta_1\right] & \text{if } j \in I_1, \\ \frac{\alpha}{2}\|\theta\|_2^2 & \text{otherwise.} \end{cases}$$

The average function $\bar{f}$ remains unchanged and the time to communicate a vector between a node of I_0 and a node of I_1 is at least $d(I_0, I_1)\tau$. Thus, the same result as Theorem 8 holds with $\Delta = d(I_0, I_1)$. Moreover, $\alpha_\ell = \alpha = \alpha_g$ and, since $m \geq n/4$, the local Lipschitz constant L_ℓ is bounded by

$$L_\ell \leq \sqrt{\frac{2n}{m}} L_g \leq 3L_g\,.$$

Replacing L_g, α_g and Δ by, respectively, $L_\ell/3$, α_ℓ and $\frac{\sqrt{2}}{3}\widetilde{\Delta}$ in Equation 7 and Equation 8 leads to the desired result.

C.2. Smooth Lower Bounds

For the smooth and convex setting, we consider the local functions of Equation 19 split on I_0 and I_1 of Lemma 24:

$$f_j(\theta) = \begin{cases} \frac{n\beta}{8m}\left[\sum_{i=1}^k (\theta_{2i+1} - \theta_{2i})^2 + \theta_1^2 - 2\theta_1\right] & \text{if } j \in I_0, \\ \frac{n\beta}{8m}\left[\sum_{i=1}^k (\theta_{2i} - \theta_{2i-1})^2 + \theta_{2k+1}^2\right] & \text{if } j \in I_1, \\ 0 & \text{otherwise.} \end{cases}$$

The average function $\bar{f}$ remains unchanged and the time to communicate a vector between a node of I_0 and a node of I_1 is at least $d(I_0, I_1)\tau$. Thus, the same result as Theorem 11 holds with $\Delta = d(I_0, I_1)$. Moreover, since $m \geq n/4$, the local smoothness β_ℓ is bounded by

$$\beta_\ell = \max_i \beta_i = \frac{n}{2m}\beta_g \leq 2\beta_g \,.$$

Replacing β_g and Δ by, respectively, $\beta_\ell/2$ and $\frac{\sqrt{2}}{3}\widetilde{\Delta}$ in Equation 9 leads to the desired result.

For the smooth and strongly convex setting, we consider the local functions of Equation 20 split on I_0 and I_1 of Lemma 24:

$$f_j(\theta) = \begin{cases} \frac{\alpha}{2}\|\theta\|_2^2 + \frac{n}{m}\frac{\beta-\alpha}{8}(\theta^\top M_1\theta - 2\theta_1) & \text{if } j \in I_0, \\ \frac{\alpha}{2}\|\theta\|_2^2 + \frac{n}{m}\frac{\beta-\alpha}{8}\theta^\top M_2\theta & \text{if } j \in I_1, \\ \frac{\alpha}{2}\|\theta\|_2^2 & \text{otherwise.} \end{cases}$$

The average function $\bar{f}$ remains unchanged and the time to communicate a vector between a node of I_0 and a node of I_1 is at least $d(I_0, I_1)\tau$. Thus, the same result as Theorem 12 holds with $\Delta = d(I_0, I_1)$. Moreover, $\alpha_\ell = \alpha = \alpha_g$ and, since $m \geq n/4$, the local condition number κ_ℓ is bounded by

$$\kappa_\ell = \frac{\max_i \beta_i}{\min_i \alpha_i} = \frac{\alpha + \frac{n}{2m}(\beta - \alpha)}{\alpha} \leq 2\kappa_g - 1 \,.$$

Replacing κ_g and Δ by, respectively, $\kappa_\ell/2$ and $\frac{\sqrt{2}}{3}\widetilde{\Delta}$ in Equation 10 leads to the desired result.

Appendix D. Proof of the Convergence Rate of MSPD (Theorem 18)

Theorem 1 (b) in Chambolle and Pock (2011) implies that, provided $\tau\sigma\lambda_1(W) < 1$, the algorithm with exact proximal step leads to a restricted primal-dual gap

$$\sup_{\|\Lambda'\|_F \leq c} \left\{\frac{1}{n}\sum_{i=1}^n f_i(\theta_i) - \operatorname{tr}\Lambda'^\top \Theta A\right\} - \inf_{\Theta' \in \mathcal{K}^n} \left\{\frac{1}{n}\sum_{i=1}^n f_i(\theta_i') - \operatorname{tr}\Lambda^\top \Theta' A\right\}$$

of

$$\varepsilon = \frac{1}{2t}\Big(\frac{nR^2}{\eta} + \frac{c^2}{\sigma}\Big).$$

This implies that our candidate Θ is such that

$$\frac{1}{n}\sum_{i=1}^{n} f_i(\theta_i) + c\|\Theta A\|_F \leq \inf_{\Theta' \in \mathcal{K}^n} \Big\{ \frac{1}{n}\sum_{i=1}^{n} f_i(\theta_i') + c\|\Theta' A\|_F + \varepsilon \Big\}.$$

Let θ be the average of all θ_i. We have:

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n} f_i(\theta) &\leq \frac{1}{n}\sum_{i=1}^{n} f_i(\theta_i) + \frac{1}{n}\sum_{i=1}^{n} L_i\|\theta_i - \theta\| \\
&\leq \frac{1}{n}\sum_{i=1}^{n} f_i(\theta_i) + \frac{1}{\sqrt{n}}\sqrt{\frac{1}{n}\sum_{i=1}^{n} L_i^2} \cdot \|\Theta(I - 11^\top/n)\|_F \\
&\leq \frac{1}{n}\sum_{i=1}^{n} f_i(\theta_i) + \frac{1}{\sqrt{n}}\sqrt{\frac{\frac{1}{n}\sum_{i=1}^{n} L_i^2}{\lambda_{n-1}(W)}} \cdot \|\Theta A\|_F.
\end{aligned}$$

Thus, if we take $c = \frac{1}{\sqrt{n}}\sqrt{\frac{\frac{1}{n}\sum_{i=1}^{n} L_i^2}{\lambda_{n-1}(W)}}$, we obtain

$$\frac{1}{n}\sum_{i=1}^{n} f_i(\theta) \leq \frac{1}{n}\sum_{i=1}^{n} f_i(\theta_*) + \varepsilon,$$

and we thus obtain an ε-minimizer of the original problem.

We have

$$\varepsilon \leq \frac{1}{2T}\Big(\frac{nR^2}{\eta} + \frac{\frac{1}{\lambda_{n-1}(W)}\frac{1}{n^2}\sum_{i=1}^{n} L_i^2}{\sigma} \Big)$$

with the constraint $\sigma\eta\lambda_1(W) < 1$. This leads to, with the choice

$$\eta = nR\sqrt{\frac{\lambda_{n-1}(W)/\lambda_1(W)}{\sum_{i=1}^{n} L_i^2/n}}$$

and taking σ to the limit $\sigma\eta\lambda_1(W) = 1$, to a convergence rate of

$$\varepsilon = \frac{1}{T}R\sqrt{\frac{1}{n}\sum_{i=1}^{n} L_i^2}\sqrt{\frac{\lambda_1(W)}{\lambda_{n-1}(W)}}\,.$$

Since we cannot use the exact proximal operator of f_i, we instead approximate it. If we approximate (with the proper notion of gap (Chambolle and Pock, 2011, Eq. (11))) each $\mathrm{argmin}_{\theta_i \in \mathcal{K}} f_i(\theta_i) + \frac{n}{2\eta}\|\theta_i - z\|^2$ up to δ_i, then the overall added gap is $\frac{1}{n}\sum_{i=1}^{n} \delta_i$. If we do M steps of subgradient descent then the associated gap is $\delta_i = \frac{L_i^2\eta}{nM}$ (standard result for strongly convex subgradient (Lacoste-Julien et al., 2012)). Therefore the added gap is

$$\frac{1}{M}R\sqrt{\frac{1}{n}\sum_{i=1}^{n} L_i^2}\sqrt{\frac{\lambda_{n-1}(W)}{\lambda_1(W)}}\,.$$

Therefore after T communication steps, i.e., communication time $T\tau$ plus MT subgradient evaluations, i.e., time MT, we get an error of

$$\Big(\frac{\widetilde{\Delta}}{T}+\frac{1}{M\widetilde{\Delta}}\Big)RL_\ell\,,$$

where $\widetilde{\Delta}=1/\sqrt{\gamma(W)}=\sqrt{\lambda_1(W)/\lambda_{n-1}(W)}$. Thus to reach ε, it takes

$$\left\lceil\frac{2RL_\ell\widetilde{\Delta}}{\varepsilon}\right\rceil\left\lceil\frac{2RL_\ell}{\varepsilon\widetilde{\Delta}}\right\rceil+\left\lceil\frac{2RL_\ell\widetilde{\Delta}}{\varepsilon}\right\rceil\tau\,,$$

which leads to the desired result.

References

Noga Alon and Vitali D. Milman. λ_1, isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, series B*, 38:73–88, 1985.

Mario Arioli and Jennifer Scott. Chebyshev acceleration of iterative refinement. *Numerical Algorithms*, 66(3):591–608, 2014.

Yossi Arjevani and Ohad Shamir. Communication complexity of distributed convex learning and optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1756–1764, 2015.

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

Winfried Auzinger. *Iterative Solution of Large Linear Systems.* Lecture notes, TU Wien, 2011.

Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE/ACM Transactions on Networking (TON)*, 14(SI):2508–2530, 2006.

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

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231–357, 2015.

Sébastien Bubeck, Qijia Jiang, Yin Tat Lee, Yuanzhi Li, and Aaron Sidford. Complexity of highly parallel non-smooth convex optimization. *arXiv e-prints*, 2019.

Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40 (1):120–145, May 2011.

John C. Duchi, Alekh Agarwal, and Martin J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic Control*, 57(3):592–606, 2012a.

John C. Duchi, Peter L. Bartlett, and Martin J. Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012b.

Darina Dvinskikh and Alexander Gasnikov. Decentralized and Parallelized Primal and Dual Accelerated Methods for Stochastic Convex Programming Problems. *arXiv e-prints*, 2019.

Lie He, An Bian, and Martin Jaggi. Cola: Decentralized linear learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4536–4546, 2018.

Niao He, Anatoli Juditsky, and Arkadi Nemirovski. Mirror prox algorithm for multi-term composite minimization and semi-separable problems. *Computational Optimization and Applications*, 61(2):275–319, 2015.

Hadrien Hendrikx, Francis Bach, and Laurent Massoulié. An accelerated decentralized stochastic proximal algorithm for finite sums. *arXiv e-prints*, 2019a.

Hadrien Hendrikx, Laurent Massoulié, and Francis Bach. Accelerated decentralized optimization with local updates for smooth and strongly convex objectives. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019b.

Martin Jaggi, Virginia Smith, Martin Takác, Jonathan Terhorst, Sanjay Krishnan, Thomas Hofmann, and Michael I Jordan. Communication-efficient distributed dual coordinate ascent. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3068–3076, 2014.

Dušan Jakovetić. A unification and generalization of exact distributed first-order methods. *IEEE Transactions on Signal and Information Processing over Networks*, 5(1):31–46, 2019.

Dušan Jakovetić, Joao Xavier, and José M. F. Moura. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59(5):1131–1146, 2014.

Dušan Jakovetić, José M. F. Moura, and Joao Xavier. Linear convergence rate of a class of distributed augmented lagrangian algorithms. *IEEE Transactions on Automatic Control*, 60(4):922–936, 2015.

Elizabeth Hou Jesús Arroyo. Efficient distributed estimation of inverse covariance matrices. In *IEEE Statistical Signal Processing Workshop (SSP)*, pages 1–5, 2016.

Zhanhong Jiang, Aditya Balu, Chinmay Hegde, and Soumik Sarkar. Collaborative deep learning in fixed topology networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5904–5914, 2017.

Anastasia Koloskova, Sebastian Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning (ICML)*, pages 3478–3487, 2019.

Simon Lacoste-Julien, Mark Schmidt, and Francis Bach. A simpler approach to obtaining an $O(1/t)$ convergence rate for the projected stochastic subgradient method. *arXiv e-prints*, 2012.

Guanghui Lan, Soomin Lee, and Yi Zhou. Communication-efficient algorithms for decentralized and stochastic optimization. *arXiv e-prints*, 2017.

Jason D. Lee, Qiang Liu, Yuekai Sun, and Jonathan E. Taylor. Communication-efficient sparse regression. *Journal of Machine Learning Research*, 18(5):1–30, 2017.

Aryan Mokhtari and Alejandro Ribeiro. DSA: Decentralized double stochastic averaging gradient algorithm. *Journal of Machine Learning Research*, 17(1):2165–2199, 2016.

Jean-Jacques Moreau. Proximité et dualité dans un espace Hilbertien. *Bulletin de la Société Mathématique de France*, 93:273–299, 1965.

Angelia Nedic and Alex Olshevsky. Distributed optimization over time-varying directed graphs. *IEEE Transactions on Automatic Control*, 60(3):601–615, 2015.

Angelia Nedic and Alex Olshevsky. Stochastic gradient-push for strongly convex functions on time-varying directed graphs. *IEEE Transactions on Automatic Control*, 61(12):3936–3947, 2016.

Angelia Nedic and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

Angelia Nedic, Alex Olshevsky, and Wei Shi. Linearly convergent decentralized consensus optimization over directed networks. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 485–489, 2016.

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

Yurii Nesterov. *Introductory Lectures on Convex Optimization : a Basic Course.* Kluwer Academic Publishers, 2004.

Shi Pu, Wei Shi, Jinming Xu, and Angelia Nedic. A push-pull gradient method for distributed optimization in networks. In *IEEE Conference on Decision and Control (CDC)*, pages 3385–3390, 2018.

Guannan Qu and Na Li. Accelerated distributed nesterov gradient descent for smooth and strongly convex functions. In *Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 209–216, 2016.

Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *International Conference on Machine Learning (ICML)*, pages 3027–3036, 2017.

Kevin Scaman, Francis Bach, Sebastien Bubeck, Laurent Massoulié, and Yin Tat Lee. Optimal algorithms for non-smooth distributed optimization in networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2740–2749, 2018.

Necdet Serhat Aybat, Alireza Fallah, Mert Gurbuzbalaban, and Asuman Ozdaglar. A Universally Optimal Multistage Accelerated Stochastic Gradient Method. *arXiv e-prints*, 2019.

Ohad Shamir. Fundamental limits of online and distributed algorithms for statistical learning and estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 163–171, 2014.

Wei Shi, Qing Ling, Kun Yuan, Gang Wu, and Wotao Yin. On the linear convergence of the ADMM in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 62(7):1750–1761, 2014.

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

Sebastian U. Stich. Local SGD converges fast and communicates little. *arXiv e-prints*, 2018.

Lu Tian and Quanquan Gu. Communication-efficient Distributed Sparse Linear Discriminant Analysis. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1178–1187, 2017.

César A. Uribe, Soomin Lee, Alexander Gasnikov, and Angelia Nedic. A dual approach for optimal algorithms in distributed optimization over networks. *arXiv e-prints*, 2018.

Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1195–1204, 2019.

Ermin Wei and Asuman Ozdaglar. Distributed alternating direction method of multipliers. In *IEEE Conference on Decision and Control (CDC)*, pages 5445–5450, 2012.

Chenguang Xi and Usman A. Khan. Dextra: A fast algorithm for optimization over directed graphs. *IEEE Transactions on Automatic Control*, 62(10):4980–4993, Oct 2017.

Ran Xin, Dušan Jakovetić, and Usman A. Khan. Distributed nesterov gradient methods over arbitrary graphs. *IEEE Signal Processing Letters*, 26(8):1247–1251, Aug 2019.