A Randomized Block Subgradient Approach to Distributed Big Data Optimization

A Randomized Block Subgradient Approach to Distributed Big Data Optimization

Francesco Farina, Giuseppe Notarstefano

***Abstract*— This paper introduces a novel distributed algorithm over static directed graphs for solving big data convex optimization problems in which the dimension of the decision variable can be extremely high and the objective function can be nonsmooth. In the proposed algorithm nodes in the network update and communicate only blocks of their current solution estimate rather than the entire vector. The algorithm consists of two main steps: a consensus step and a subgradient update on a single block of the optimization variable (which is then broadcast to neighbors). Agents are shown to asymptotically achieve consensus by studying a block-wise consensus protocol over random graphs. Then convergence to the optimal cost is proven in expected value by exploiting the consensus of agents estimates and randomness of the algorithm. Finally, as a numerical example, a distributed linear classification problem is solved by means of the proposed algorithm.**

I. Introduction

Distributed coordination and control over networks have gained increasing attention in recent years. In fact, many problems arising in this scenario can be formulated as optimization problems which need to be solved in a cooperative way. In this distributed set-up agents in the network typically do not know the entire optimization problem, but rather perform local computations based on local objective functions (and constraints) and communicate with neighboring agents only. Many distributed optimization algorithms have been proposed in recent years, some of which involving nonsmooth objective functions. In such cases subgradient-based algorithms have been designed. The first algorithms of this type appeared in [1, 2], while recent advances involve more sophisticated protocols, such as push-sum, to deal with directed communication [3]. Recently, extensions to the stochastic setting have also appeared, e.g., a stochastic distributed mirror descent was proposed in [4]. Finally, distributed algorithms over random networks are relevant for this paper. In [5], consensus protocols were studied using random row-stochastic matrices, while in [6] a distributed subgradient method over random networks with underlying doubly stochastic matrices has been proposed.

Applying the above algorithms to big data problems may be hard to implement since many problems related to the high dimension of the decision variable may arise. For example, a limited communication bandwidth may not allow nodes to communicate the entire solution estimate. Thus, there is the need to develop tailored distributed algorithms for *big data* optimization problems in which only few blocks of the entire (local) solution estimate are exchanged among the agents.

Block coordinate methods have a long history in the centralized optimization literature (see, e.g., [7] for a survey). They have been originally designed for solving smooth problems. However, an increasing number of results for dealing with nonsmooth objective functions have started to appear in the last years. Blocks to be updated can be chosen in various ways: cyclically, almost cyclically [8] or randomly. In the last case, a randomized block coordinate descent algorithm has been proposed in [9], while in [10] a stochastic block mirror descent method with random block update is proposed. Also parallel block coordinate methods have been widely studied in the optimization literature (see, e.g., [11]). Problems involving smooth convex functions are addressed in [12], while composite optimization problems are studied in [13, 14]. Moreover, [15] proposed a unified framework for nonsmooth optimization using block algorithms, treating both the centralized and the parallel case. Distributed algorithms which are capable to deal with block communication started to be studied only recently. In [16], nonconvex problems with nonsmooth regularizers are addressed by means of a block gradient tracking scheme, while in [17] an asynchronous algorithm for nonconvex optimization based on the method of multipliers, which is implementable block-wise have been proposed. Smooth problems with common cost function and linear constraints has been addressed in [18] through a randomized block algorithm.

In this paper, we propose a distributed algorithm for solving (over networks) big data convex optimization problems with nonsmooth objective function, in which, at each iteration, each node exchanges with its neighbors only a single block of the optimization variable. The communication network is modeled as a directed graph admitting a doubly stochastic weight matrix. At each iteration each agent performs a consensus step, computes a subgradient at the computed point, and performs a subgradient update on a randomly chosen block only. Then it broadcasts to its neighbors only the updated block. The local block-wise subgradient updates and the communication of a single block of the decision variable at each iteration require nontrivial analysis to show the convergence of the algorithm. In fact, it is worth noting that, despite the double stochasticity of the weight matrix, the consensus step on each block turns out to be performed using a sequence of random

The authors are with the Department of Electrical, Electronic and Information Engineering "G. Marconi", Università di Bologna, Bologna, Italy. email: {franc.farina,giuseppe.notarstefano}@unibo.it

This result is part of a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 638992 - OPT4SMART).

row-stochastic matrices. The analysis of the algorithm is carried out in two parts. At first, it is shown that the agents in the network asymptotically reach consensus in expected value by studying block-wise, perturbed consensus dynamics with random matrices. Then, convergence to the optimal cost in expected value is proven by properly bounding errors due to the block-wise update and exploiting the property that blocks are uniformly drawn. The results presented in this paper can be extended to deal with a more general problem set-up involving stochastic objective functions and constraints. This extension is the subject of an extended work in which the complete convergence proofs will be provided for a more general class of algorithms and for the extended set-up.

The paper is organized as follows. The considered problem is introduced in Section II along with some preliminary results. In Section III, the algorithm is presented and, then, analyzed in Section IV. A numerical example is provided in Section V and, finally, some conclusions are drawn in Section VI.

II. Set-up and preliminaries

A. Distributed optimization set-up

Consider the following unconstrained optimization problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad f(x)=\sum_{i=1}^{N} f_i(x) \tag{1}$$

where the functions $f_i:\mathbb{R}^n\to\mathbb{R}$, $n\gg 1$, are convex and possibly non smooth. We denote by $x^\star\in\mathbb{R}^n$ a solution of problem (1). The optimization variable $x\in\mathbb{R}^n$ has a block structure, i.e.,

$$x=[x[1]^\top,\dots,x[B]^\top]^\top,$$

where we have defined by $x[\ell]$ the ℓ-th block of x. Namely, given a partition of the identity matrix $U=[U^1,\dots,U^B]$, with $U^\ell\in\mathbb{R}^{n\times n_\ell}$ for all i and $\sum_{\ell=1}^{B} n_\ell=n$, we can retrieve $x=\sum_{i=1}^{B} U^\ell x[\ell]$ and $x[\ell]=(U^\ell)^\top x$. Let us define $g_i(x)\in\partial f_i(x)$ as a subgradient of f_i computed at x. Similarly, $g(x)\in\partial f(x)$ denotes a subgradient of f computed at x. Then, we make the following assumption on the functions f_i.

Assumption 1 (Bounded subgradients)**.** Given problem (1), there exist constants $G_i\in[0,\infty)$ such that $\|g_i(x)\|\le G_i$, $\forall x$ and for all $i=1,\dots,N$. □

Notice that, Assumption 1 implies that $\|g_i(x)[\ell]\|\le G_i$ for all ℓ. Moreover $\|g(x)\|\le G=\sum_{i=1}^{N} G_i$ and hence $\|g_i(x)\|\le G$ for all i.

Problem (1) is to be solved in a distributed way by a network of N agents. The network is modeled through a weighted *strongly connected* directed graph $\mathcal{G}=(\mathcal{V},\mathcal{E},\mathcal{W})$ with $\mathcal{V}=\{1,\dots,N\}$, $\mathcal{E}\subseteq\mathcal{V}\times\mathcal{V}$ and $\mathcal{W}\in\mathbb{R}^{N\times N}$ being a weight matrix. Each agent knows only a portion of the entire problem, namely agent i is assigned the function f_i. We denote by $\mathcal{N}_i^{out}$ the set of out-neighbors of node i, i.e., $\mathcal{N}_i^{out}=\{j\mid(i,j)\in\mathcal{E}\}$. Similarly, the set of in-neighbors of node i, is defined as $\mathcal{N}_i^{in}=\{j\mid(j,i)\in\mathcal{E}\}$. We say that a directed graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ contains a spanning tree if for some $v\in\mathcal{V}$ there exists a directed path from the vertex v to all other vertices $u\in\mathcal{V}$. The following assumption holds on the weight matrix $\mathcal{W}$.

Assumption 2 (Doubly stochastic matrix)**.** For all $i,j\in\{1,\dots,N\}$, the weights w_{ij} of the weight matrix $\mathcal{W}$ satisfy

(i) $w_{ii}>0$;
(ii) if $i\neq j$, $w_{ij}>0$ if and only if $j\in\mathcal{N}_i^{in}$;
(iii) there exists a constant $\eta>0$ such that $w_{ii}\ge\eta$ and if $w_{ij}>0$, then $w_{ij}\ge\eta$;
(iv) $\sum_{i=1}^{N} w_{ij}=1$ and $\sum_{j=1}^{N} w_{ij}=1$. □

B. Preliminary results

Consider a stochastic, discrete-time dynamical system evolving according to

$$x^{t+1}=A^t x^t,\quad\forall t, \tag{2}$$

where $\{A^t\}_{t\ge0}$ is a sequence of random $n\times n$ row-stochastic matrices. Let $(\Omega,\mathcal{F},P)$ be a probability space, we assume the sequence $\{A^t,\mathcal{S}^t\}_{t\ge0}$ form an *adapted process*, i.e., $\{A^t\}_{t\ge0}$ is a stochastic process defined on $(\Omega,\mathcal{F},P)$, $\{\mathcal{S}^t\}_{t\ge0}$ is a filtration (i.e., $\mathcal{S}^t\subseteq\mathcal{S}^{t+1}$ and $\mathcal{S}^t\in\mathcal{F}$ for all t) and A^t is measurable with respect to $\mathcal{S}^t$. Given a sequence of matrices $\{A^t\}_{t\ge0}$, let us define the transition matrix from iteration s to iteration t as

$$\Phi(t,s)=\begin{cases}A^tA^{t-1}\dots A^s, & \text{if } t>s,\\ A^t, & \text{if } t=s.\end{cases}$$

Given a nonnegative matrix A and some $\delta\in(0,1)$, we denote by A_δ the matrix whose entries are defined as

$$[A_\delta]_{ij}=\begin{cases}\delta, & \text{if } A_{ij}\ge\delta,\\ 0, & \text{otherwise.}\end{cases}$$

We say that A contains a δ-spanning tree if the graph induced by A_δ contains a spanning tree. Moreover, given a vector $x\in\mathbb{R}^n$, we define $d(x)=\max_{1\le i\le n}x[i]-\min_{1\le i\le n}x[i]$. Then, the following result holds true for system (2).

Lemma 1 ([5, Theorem 3.1])**.** *Consider system* (2)*. If there exist $h>0$, $\delta>0$ such that $\mathbf{E}[\sum_{t=mh+1}^{(m+1)h}A^t\mid\mathcal{S}^{mh}]$ contains a δ-spanning tree for each m and $A^t\ge\delta I$ for each t, then, for any x^0 such that $\mathbf{E}[\|x^0\|^p]<\infty$ (which is independent of $\{A^t\}_{t\ge0}$), and any $p>0$, it holds*

$$\begin{aligned}\mathbf{E}[d(x^t)^p]&=\mathbf{E}[d(\Phi(t,0)x^0)^p]\\&\le\mu^t\mathbf{E}[d(x^0)^p]\le M\mu^t\mathbf{E}[\|x^0\|^p],\end{aligned}$$

where $M\in(0,\infty)$ and $\mu\in(0,1)$. □

The following corollary specializes the above result to the case of i.i.d. random matrices whose expected value is a doubly stochastic matrix.

Corollary 1. *Consider the discrete system* (2)*. Let $\{A^t\}_{t\ge0}$ be a sequence of independent and identically distributed random row-stochastic matrices, such that $A^t\ge\delta I$ for all t for some $\delta>0$, and $\mathbf{E}[A^t]$ is a doubly stochastic matrix. Then,*

for any x^0 such that $\mathbf{E}[\|x^0\|^p] < \infty$ (which is independent of $\{A^t\}_{t\geq 0}$), and any $p > 0$, it holds

$$\begin{aligned}\mathbf{E}[d(x^t)^p] &= \mathbf{E}[d(\Phi(t,0)x^0)^p] \\ &\leq \mu^t \mathbf{E}[d(x^0)^p] \leq M\mu^t \mathbf{E}[\|x^0\|^p],\end{aligned}$$

where $M \in (0,\infty)$ and $\mu \in (0,1)$. □

Finally, the following two results will be useful in the rest of the paper.

Lemma 2 ([19, Lemma 3.1])**.** *Let $\{\gamma_t\}_{t\geq 0}$ be a scalar sequence.*

1) *If $\lim_{t\to\infty} \gamma_t = \xi$ and $\beta \in (0,1)$ then $\lim_{t\to\infty} \sum_{s=0}^{t} \beta^{t-s}\gamma_s = \frac{\xi}{1-\beta}$.*
2) *If $\gamma_t \geq 0$ for all t, $\sum_{t=0}^{\infty} \gamma_t < \infty$ and $\beta \in (0,1)$, then $\sum_{t=0}^{\infty} \left(\sum_{s=0}^{t} \beta^{t-s}\gamma_s\right) < \infty$.*

□

Lemma 3 (Tower property of conditional expectation)**.** *Let X be a random variable defined on a probability space $(\Omega, \mathcal{F}, P)$. Let $Z \subseteq Y \subseteq \mathcal{F}$. Then, $\mathbf{E}[\mathbf{E}[X \mid Y, Z] \mid Z] = \mathbf{E}[X \mid Z]$.* □

III. Distributed Block Subgradient Algorithm

The Distributed Block Subgradient method for solving problem (1) in a distributed way is now presented. The algorithm works as follow. Each agent i maintains a local solution estimate x_i^t, which is initialized at x_i^0. We assume that the initial estimates are entirely shared among neighboring agents. Then, at each iteration t, agent i performs two updates:

(i) it computes a weighted average of its in-neighbors' estimates x_j^t, $j \in \mathcal{N}_i^{in}$ and stores it in y_i^{t+1};
(ii) it picks a random block $\ell_i^t \in \{1,\dots,B\}$ (with uniform probability) and computes x_i^{t+1} by updating the ℓ_i^t-th block of x_i^t by using the ℓ_i^t-th block of a subgradient of f_i computed at y_i^{t+1} and leaving the other blocks unchanged.

Finally, it broadcasts $x_i^{t+1}[\ell_i^t]$ to its out-neighbors. A pseudocode of the method is reported in Algorithm 1.

Algorithm 1 Distributed Block Subgradient

Initialization: x_i^0

Evolution: for $t = 0, 1, \dots$

UPDATE

$$y_i^{t+1} = \sum_{j\in\mathcal{N}_i^{in}} w_{ij} x_j^t \tag{3}$$

PICK $\ell_i^t \in \{1,\dots,B\}$ with $P(\ell_i^t = \ell) = \frac{1}{B}$, $\forall \ell$

UPDATE

$$x_i^{t+1}[\ell] = \begin{cases} y_i^{t+1}[\ell] - \alpha_t g_i(y_i^{t+1})[\ell] & \text{if } \ell = \ell_i^t \\ x_i^t[\ell] & \text{else} \end{cases} \tag{4}$$

BROADCAST $x_i^{t+1}[\ell_i^t]$ to $j \in \mathcal{N}_i^{out}$

It is worth noting that, although node i receives from each $j \in \mathcal{N}_i^{in}$ only block $x_j^t[\ell_i^{t-1}]$, the consensus step (3) can be performed using the entire x_j^t since the other blocks have not changed since the last time they have been received.

As for the block-wise subgradient update (4), notice that the ℓ_i^t-th block of the *whole* subgradient computed at y_i^{t+1} is used. This is due to the fact that computing a subgradient with respect to the ℓ_i^t-th component only is, in general, *not* equivalent to picking the ℓ_i^t-th block of the *whole* subgradient. This will turn out to be a fundamental property in the algorithm analysis. There are two cases in which the ℓ_i^t-th block of the subgradient can be directly computed. If functions f_i are separable on the blocks, then the ℓ_i^t-th block of the (sub)gradient can be directly computed as the (sub)gradient with respect to that block. Also, if functions f_i are smooth, the same happens. In these cases, the algorithm can be further simplified.

The last important feature of the Distributed Block Subgradient algorithm that we want to highlight involves consensus step (3). Define z_ℓ^t as the vector stacking the ℓ-th component of all the x_i^t, i.e., $z_\ell^t = [(x_1^t[\ell])^\top, \dots, (x_N^t[\ell])^\top]^\top$. Define the matrix $\mathcal{U}_\ell^t$ as a diagonal matrix in which the i-th element of the diagonal is set to 1 if $\ell_i^t = \ell$ and it is set to 0 otherwise, i.e.

$$[\mathcal{U}_\ell^t]_{ij} = \begin{cases} 1, & \text{if } i = j \text{ and } \ell = \ell_i^t, \\ 0, & \text{otherwise.} \end{cases}$$

Finally, let $\mathcal{U}_{-\ell}^t = I - \mathcal{U}_\ell^t$. Now, consider a consensus protocol associated to the update (3) in Algorithm 1, i.e., a system evolving according to

$$x_i^{t+1} = \sum_{j\in\mathcal{N}_i^{in}} w_{ij} x_j^t, \tag{5}$$

for all i and t. Then, (5) can be rewritten in terms of z_ℓ as

$$z_\ell^{t+1} = \tilde{W}_\ell^t z_\ell^t,$$

where $\tilde{W}_\ell^t = \mathcal{U}_{-\ell}^t + \mathcal{U}_\ell^t \mathcal{W}$. It can be easily seen that, for all ℓ and t, the matrix $\tilde{W}_\ell^t$ is row-stochastic but, in general, no more doubly-stochastic.

Remark 1. In [16] a block-wise dynamic push-sum consensus algorithm has been used, which guarantees average tracking. In this paper, we are not designing a block-wise average tracking scheme, but just to a protocol guaranteeing convergence of the optimization scheme. Thus, we are able to deal with row-stochastic matrices without resorting to push-sum protocols, by building on properties of sequences of random row-stochastic matrices [5]. □

IV. Algorithm Convergence

In this section, the convergence of the Distributed Block Subgradient algorithm is proven. The proof is split in two parts: the first showing consensus of the agents' solution estimates, and the second proving convergence to the optimal cost. In both cases, convergence is to be intended in expected value.

The following two assumptions are required for the stepsize sequence and the random variables ℓ_i^t.

Assumption 3 (Diminishing stepsize)**.** The sequence $\{\alpha_t\}$ satisfies

$$\sum_{t=0}^{\infty} \alpha_t = \infty, \qquad \sum_{t=0}^{\infty} \alpha_t^2 < \infty.$$

Moreover, $\alpha_{t+1} \leq \alpha_t$ for all t, with $\alpha_{-1} = \alpha_0$. □

Assumption 4 (i.i.d. draws)**.** The random variables ℓ_i^t are independent and identically distributed for all $i = 1, \dots, N$.

A. Consensus

In this section it is shown that the sequences $\{x_i^t\}_{t\geq 0}$ and $\{y_i^t\}_{t\geq 0}$ computed by each agent in the network asymptotically achieve consensus in expected value.

Let us start by defining $\boldsymbol{x}^t = [(x_1^t)^\top, \dots, (x_N^t)^\top]^\top$ and

$$\bar{x}^t = \frac{1}{N} \sum_{i=1}^{N} x_i^t. \tag{6}$$

Moreover, we define $\mathcal{S}_t$ as the state of the network at iteration t and $\mathcal{S}_{[t]} = [\mathcal{S}_t, \mathcal{S}_{t-1}, \dots, \mathcal{S}_0]$.

Then, the following result provides a bound on the expected distance between x_i^t and $\bar{x}^t$ at iteration t, conditioned to $\mathcal{S}_0$.

Lemma 4. *Let Assumptions 1, 2 and 4 hold. Then, there exist constants $M \in (0, \infty)$ and $\mu_M \in (0, 1)$ such that*

$$\mathbf{E}[\|x_i^t - \bar{x}^t\| \mid \mathcal{S}_0] \leq MB \left(\mu_M^{t-1} \|\boldsymbol{x}^0\| + G \sum_{s=0}^{t-2} \mu_M^{t-s-2} \alpha_s + G\alpha_{t-1} \right) \tag{7}$$

for all $i \in \{1, \dots, N\}$ and all t.

Proof. For the sake of space we provide only a sketch of the proof with the main steps and leave all the derivations to a forthcoming document in which the convergence of a more general class of algorithms is proved for a more general optimization set-up. Assume for simplicity that the number of blocks is equal to the dimension of the optimization variable, i.e., $B = n$. Moreover, define $g_\ell^t = [g_1(y_1^{t+1})[\ell], \dots, g_N(y_N^{t+1})[\ell]]^\top$. Now, Algorithm 1 can be rewritten with respect to z_ℓ as

$$z_\ell^{t+1} = \tilde{W}_\ell^t z_\ell^t + e_t, \tag{8}$$

where $\tilde{W}_\ell^t = \mathcal{U}_{-\ell}^t + \mathcal{U}_\ell^t \mathcal{W}$ and $e_t = -\alpha_t \tilde{g}_\ell^t$, with $\tilde{g}_\ell^t = \mathcal{U}_\ell^t g_\ell^t$. By building on the structure of the matrices $\mathcal{U}_\ell^t$ and $\mathcal{U}_{-\ell}^t$, the expected value of $\tilde{W}_\ell^t$ can shown to be, for all t,

$$\mathbf{E}[\tilde{W}_\ell^t] = \frac{B-1}{B} U + \frac{1}{B} \mathcal{W}$$

which is doubly stochastic and such that $\mathbf{E}[\tilde{W}_\ell^t] > \frac{1}{B}\eta I$ (see Assumption 2). Thanks to this, it can be shown that by combining (8) with Corollary 1 and Assumption 1, one has

$$\mathbf{E}[d(z_\ell^{t+1}) \mid \mathcal{S}_0] \leq M \left(\mu_\ell^t \|z_\ell^0\| + G \sum_{s=0}^{t-1} \mu_\ell^{t-s-1} \alpha_s + G\alpha_t \right).$$

The derivation is omitted for the sake of space. The proof can be completed by defining $\bar{z}_\ell^t = \frac{1}{N} \sum_{i=1}^N z_\ell^t[i]$ and by noticing that $|z_\ell^t[j] - \bar{z}_\ell^t| \leq \max_i z_\ell^t[i] - \min_i z_\ell^t[i]$, for all $j \in \{1, \dots, N\}$, and, by definition, $x_i^t[\ell] = z_\ell^t[i]$ and $\bar{x}^t[\ell] = \bar{z}_\ell^t$. ■

The following lemma is a direct consequence of Lemma 4 under Assumption 3 and it shows that asymptotic consensus is achieved.

Lemma 5. *Let Assumptions 1, 2, 3 and 4 hold. Then,*

$$\lim_{t\to\infty} \mathbf{E}[\|x_i^t - \bar{x}^t\| \mid \mathcal{S}_0] = 0$$

for all $i \in \{1, \dots, N\}$. □

We conclude this section by providing three results that will be used to prove the convergence of the algorithm.

By using Lemma 4, the expected value of the distance between y_i^t and x_i^t can be bounded, by exploiting the convexity of the norm.

Lemma 6. *Let Assumptions 1, 2 and 4 hold. Then,*

$$\mathbf{E}[\|y_i^{t+1} - x_i^t\| \mid \mathcal{S}_0] \leq 2\mathbf{E}[\|x_i^t - \bar{x}^t\| \mid \mathcal{S}_0] \tag{9}$$

for all $i \in \{1, \dots, N\}$ and all t. □

Moreover, the following two results show that the series $\sum_{\tau=0}^{\infty} \alpha_\tau \mathbf{E}[\|x_i^\tau - \bar{x}^\tau\|]$ and $\sum_{\tau=0}^{\infty} \alpha_\tau \mathbf{E}[\|y_i^{\tau+1} - x_i^\tau\|]$ are summable for all $i \in \{1, \dots, N\}$.

Lemma 7. *Let Assumptions 1, 2, 3 and 4 hold. Then,*

$$\lim_{t\to\infty} \sum_{\tau=0}^{t} \alpha_\tau \mathbf{E}[\|x_i^\tau - \bar{x}^\tau\| \mid \mathcal{S}_0] < \infty$$

for all $i \in \{1, \dots, N\}$.

Proof. The proof is based on using Lemma 4 and the fact that, by Assumption 3, one has $\alpha_{t+1} \leq \alpha_t$. By doing so it can be shown that

$$\begin{aligned} \sum_{\tau=0}^{t} \alpha_\tau \mathbf{E}[\|x_i^\tau - \bar{x}^\tau\| \mid \mathcal{S}_0] \leq MB \Bigg(& \|x^0\| \sum_{\tau=0}^{t} \mu_M^{\tau-1} \alpha_\tau \\ & + G \sum_{\tau=0}^{t} \sum_{s=0}^{\tau-2} \mu_M^{\tau-s-2} \alpha_s^2 \\ & + G \sum_{\tau=0}^{t} \alpha_{\tau-1}^2 \Bigg). \end{aligned}$$

which leads to (9), by using Assumption 3 and Lemma 2. ■

Corollary 2. *Let Assumptions 1, 2, 3 and 4 hold. Then,*

$$\lim_{t\to\infty} \sum_{\tau=0}^{t} \alpha_\tau \mathbf{E}[\|y_i^{\tau+1} - x_i^\tau\| \mid \mathcal{S}_0] < \infty$$

for all $i \in \{1, \dots, N\}$. □

B. Optimality

The main result of this paper is provided in this section. A bound on the expected distance from the optimal cost at iteration t is given. Moreover, it is shown that such a distance goes to 0 as $t \to \infty$.

Theorem 1. *Let Assumptions 1, 2, 3 and 4 hold. Then,*

$$\begin{aligned}\min_{\tau\le t}&(\mathbf{E}[f(\bar{x}^\tau)\mid\mathcal{S}_0]-f(x^\star))\\ &\le\left(\sum_{\tau=0}^{t}\alpha_\tau\right)^{-1}\left(\frac{B}{2}\sum_{j=1}^{N}\|x_j^0-x^\star\|^2+\sum_{\tau=0}^{t}\frac{\alpha_\tau^2NBG^2}{2}\right.\\ &\quad+\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}G_j\mathbf{E}[\|y_j^{\tau+1}-x_j^\tau\|\mid\mathcal{S}^0]\\ &\quad\left.+\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}G_j\mathbf{E}[\|x_j^\tau-\bar{x}^\tau\|\mid\mathcal{S}^0]\right).\end{aligned}\tag{10}$$

Moreover,

$$\lim_{t\to\infty}\min_{\tau\le t}\left(\mathbf{E}\left[f(\bar{x}^\tau)\mid\mathcal{S}_0\right]-f(x^\star)\right)=0.\tag{11}$$

Proof. For the sake of space we provide only a sketch of the proof with the main steps and leave all the derivations to a forthcoming document in which the convergence of a more general class of algorithms is proved for a more general optimization set-up. From the convexity of f we have that, at a given iteration t,

$$\begin{aligned}\left(\sum_{\tau=0}^{t}\alpha_\tau\right)&\min_{\tau\le t}(\mathbf{E}[f(\bar{x}^\tau)\mid\mathcal{S}_0]-f(x^\star))\\ &\le\sum_{\tau=0}^{t}\alpha_\tau\left(\mathbf{E}[f(\bar{x}^\tau)\mid\mathcal{S}_0]-f(x^\star)\right).\end{aligned}\tag{12}$$

$$\begin{aligned}&\overset{(a)}{\le}\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}\left(\mathbf{E}[f_j(y_j^{\tau+1})\mid\mathcal{S}_0]-f_j(x^\star)\right)\\ &\quad+\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}G_j\mathbf{E}[\|y_j^{\tau+1}-x_j^\tau\|\mid\mathcal{S}_0]\\ &\quad+\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}G_j\mathbf{E}[\|x_j^\tau-\bar{x}^\tau\|\mid\mathcal{S}_0].\end{aligned}\tag{13}$$

where (a) can be shown by using the convexity of f and the subgradient boundedness Assumption 1. We do not report the steps to show (a) for the sake of space.

The rest of the proof is mainly base on bounding the term $\sum_{\tau=0}^{t}\alpha_\tau\sum_{j=1}^{N}\left(\mathbf{E}[f_j(y_j^{\tau+1})\mid\mathcal{S}_0]-f_j(x^\star)\right)$ in the above inequality. In order to simplify the notation, let us denote by $g_i^\tau=g_i(y_i^{\tau+1})$ and by $g_{i,\ell_i^\tau}=g_i(y_i^{\tau+1})[\ell_i^\tau]$. From (4), one has

$$\begin{aligned}\|x_i^{\tau+1}[\ell_i^\tau]-x^\star[\ell_i^\tau]\|^2&=\|y_i^{\tau+1}[\ell_i^\tau]-\alpha_\tau g_{i,\ell_i^\tau}-x^\star[\ell_i^\tau]\|^2\\ &=\|y_i^{\tau+1}[\ell_i^\tau]-x^\star[\ell_i^\tau]\|^2\\ &\quad-2\alpha_\tau\langle U^{\ell_i^\tau}g_i^\tau,y_i^{\tau+1}-x^\star\rangle\\ &\quad+\alpha_\tau^2\|g_{i,\ell_i^\tau}\|^2.\end{aligned}\tag{14}$$

Hence, by definition,

$$\|x_i^{\tau+1}[\ell]-x^\star[\ell]\|^2=\begin{cases}(14), & \text{if } \ell=\ell_i^\tau,\\ \|x_i^\tau[\ell]-x^\star[\ell]\|^2, & \text{otherwise.}\end{cases}\tag{15}$$

Now, by taking the expected value conditioned to $\mathcal{S}_{[\tau]}$, and using the fact that $\sum_{\ell=1}^{B}U^\ell g_i^\tau=g_i^\tau$ and $\|g_{i,\ell}\|\le G_i$ for all ℓ, it can be shown that

$$\begin{aligned}\mathbf{E}&[\|x_i^{\tau+1}-x^\star\|^2\mid\mathcal{S}_{[\tau]}]\\ &\le\frac{B-1}{B}\|x_i^\tau-x^\star\|^2+\frac{1}{B}\|y_i^{\tau+1}-x^\star\|^2\\ &\quad-2\frac{\alpha_\tau}{B}\langle g_i^\tau,y_i^{\tau+1}-x^\star\rangle+\alpha_\tau^2G_i^2.\end{aligned}\tag{16}$$

By summing over i, one has

$$\begin{aligned}\sum_{i=1}^{N}\mathbf{E}[\|x_i^{\tau+1}-x^\star\|^2\mid\mathcal{S}_{[\tau]}]&\le\sum_{i=1}^{N}\|x_i^\tau-x^\star\|^2+\alpha_\tau^2NG^2\\ &\quad-2\frac{\alpha_\tau}{B}\sum_{i=1}^{N}\left(f_i(y_i^{\tau+1})-f_i(x^\star)\right),\end{aligned}\tag{17}$$

where we do not report the derivation of the above condition for the sake of space. It requires to use the definition of $y_i^{\tau+1}$ and the double stochasticity of W. Now, by using Lemma 3, one gets

$$\begin{aligned}&2\sum_{i=1}^{N}\frac{\alpha_\tau}{B}\left(\mathbf{E}[f_i(y_i^{\tau+1})\mid\mathcal{S}_0]-f_i(x^\star)\right)\\ &\le\sum_{i=1}^{N}\mathbf{E}[\|x_i^\tau-x^\star\|^2\mid\mathcal{S}_0]-\sum_{i=1}^{N}\mathbf{E}[\|x_i^{\tau+1}-x^\star\|^2\mid\mathcal{S}_0]\\ &\quad+\alpha_\tau^2NG^2.\end{aligned}\tag{18}$$

which, by summing over τ, and noting that the previous relation is a telescopic series, leads to

$$\begin{aligned}\sum_{\tau=0}^{t}\alpha_\tau\sum_{i=1}^{N}&\left(\mathbf{E}[f_i(y_i^{\tau+1})\mid\mathcal{S}_0]-f_i(x^\star)\right)\\ &\le\frac{B}{2}\sum_{i=1}^{N}\|x_i^0-x^\star\|^2+\sum_{\tau=0}^{t}\frac{\alpha_\tau^2NBG^2}{2}.\end{aligned}\tag{19}$$

By combining (13) and (19) one obtains (10). The proof can be completed by taking the limit for $t\to\infty$ and using Lemma 7 and Corollary 2 in (10). ■

The following corollary follows immediatelly from Lemma 5.

Corollary 3. *Let Assumptions 1, 2, 3 and 4 hold. Then,*

$$\lim_{t\to\infty}\min_{\tau\le t}\left(\mathbf{E}\left[f(x_i^\tau)\mid\mathcal{S}_0\right]-f(x^\star)\right)=0\tag{20}$$

for all $i\in\{1,\dots,N\}$.

V. Numerical example

Consider a network of N agents. Each agent $i\in\{1,\dots,N\}$ has m_i training samples $a_{i,1},\dots,a_{i,m_i}\in\mathbb{R}^d$ each of which is associated a binary label $b_{i,j}\in\{-1,1\}$ for all $j\in\{1,\dots,m_i\}$. The goal of the network is to build a linear classifier from the training samples, i.e., to find a hyperplane of the form $\{z\in\mathbb{R}^d\mid\langle\theta,z\rangle+\theta_0=0\}$ where $\theta\in\mathbb{R}^d$ and $\theta_0\in\mathbb{R}$. Let us define $x=[\theta^\top,\theta_0]^\top\in\mathbb{R}^{d+1}$

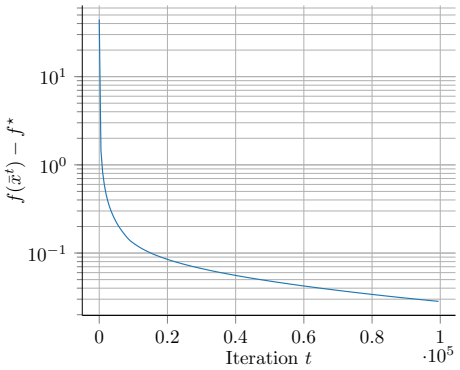

Fig. 1: Evolution of the cost error.

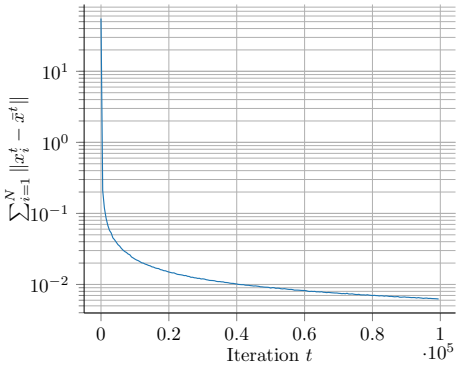

Fig. 2: Evolution of the consensus error.

and $\hat{a}_{i,j} = [a_{i,j}^\top, 1]^\top$. Then, the solution of such problem can be determined by solving the following convex optimization problem in which a regularized logistic loss is used as cost function

$$\underset{x\in\mathbb{R}^{d+1}}{\text{minimize}} \quad \sum_{i=1}^{N}\sum_{j=1}^{m_i} \log\left(1+\exp(-b_{i,j}\langle x, \hat{a}_{i,j}\rangle)\right) + \lambda\|x\|_1, \tag{21}$$

where $\lambda > 0$ is the regularization weight. The behavior of the Distributed Block Subgradient method is tested in this scenario with $N = 20$ agents, $x \in \mathbb{R}^{11}$ and $B = 11$ (i.e., one block per coordinate). A synthetic dataset of 200 points belonging to two different (non separable) clusters has been created and 10 points has been assigned to each agent, i.e. $m_1 = \cdots = m_N = 10$. Agents communicate according to an undirected, connected graph generated according to an Erdős-Rènyi random model with connectivity parameter $p = 0.2$. The corresponding weight matrix is built by using the Metropolis-Hastings rule. Finally, we set $\lambda = 1$ and the stepsize $\alpha_t = \frac{0.1}{t^{0.51}}$. The evolution of the cost and the consensus errors is reported in Figure 1 and Figure 2 respectively.

VI. Conclusions

In this paper, we presented the Distributed Block Subgradient algorithm for solving, in a distributed fashion, big data convex optimization problems in which the dimension of the decision variable is very high and the cost function may be nonsmooth. The algorithm is particularly well suited for big data optimization problems since agents in the network can communicate a single block of the optimization variable per iteration. It is shown that the agents in the network asymptotically agree on a common solution which is shown to be cost-optimal in expected value. As a numerical example, the Distributed Block Subgradient algorithm is tested on a linear classification problem with regularized logistic loss.

References

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

[2] B. Johansson, M. Rabi, and M. Johansson, "A randomized incremental subgradient method for distributed optimization in networked systems," *SIAM Journal on Optimization*, vol. 20, no. 3, pp. 1157–1170, 2009.

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

[4] J. Li, G. Li, Z. Wu, and C. Wu, "Stochastic mirror descent method for distributed multi-agent optimization," *Optimization Letters*, vol. 12, no. 6, pp. 1179–1197, 2018.

[5] B. Liu, W. Lu, and T. Chen, "Consensus in networks of multiagents with switching topologies modeled as adapted stochastic processes," *SIAM Journal on Control and Optimization*, vol. 49, no. 1, pp. 227–253, 2011.

[6] I. Lobel and A. Ozdaglar, "Distributed subgradient methods for convex optimization over random networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1291–1306, 2011.

[7] A. Beck and L. Tetruashvili, "On the convergence of block coordinate descent type methods," *SIAM journal on Optimization*, vol. 23, no. 4, pp. 2037–2060, 2013.

[8] Y. Xu and W. Yin, "A globally convergent algorithm for nonconvex optimization based on block coordinate update," *Journal of Scientific Computing*, vol. 72, no. 2, pp. 700–734, 2017.

[9] P. Richtárik and M. Takáč, "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," *Mathematical Programming*, vol. 144, no. 1-2, pp. 1–38, 2014.

[10] C. D. Dang and G. Lan, "Stochastic block mirror descent methods for nonsmooth and stochastic optimization," *SIAM Journal on Optimization*, vol. 25, no. 2, pp. 856–881, 2015.

[11] S. J. Wright, "Coordinate descent algorithms," *Mathematical Programming*, vol. 151, no. 1, pp. 3–34, 2015.

[12] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[13] P. Richtárik and M. Takáč, "Parallel coordinate descent methods for big data optimization," *Mathematical Programming*, vol. 156, no. 1-2, pp. 433–484, 2016.

[14] I. Necoara and D. Clipici, "Parallel random coordinate descent method for composite minimization: Convergence analysis and error bounds," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 197–226, 2016.

[15] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013.

[16] I. Notarnicola, Y. Sun, G. Scutari, and G. Notarstefano, "Distributed big-data optimization via block-iterative gradient tracking," *arXiv preprint arXiv:1808.07252*, 2018.

[17] F. Farina, A. Garulli, A. Giannitrapani, and G. Notarstefano, "A distributed asynchronous method of multipliers for constrained nonconvex optimization," *Automatica*, vol. 103, pp. 243 – 253, 2019.

[18] I. Necoara, "Random coordinate descent algorithms for multi-agent convex optimization over networks," *IEEE Transactions on Automatic Control*, vol. 58, no. 8, pp. 2001–2012, 2013.

[19] 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.