2019 IEEE 58th Conference on Decision and Control (CDC)
Palais des Congrès et des Expositions Nice Acropolis
Nice, France, December 11-13, 2019

On Primal and Dual Approaches for Distributed Stochastic Convex Optimization over Networks

Darina Dvinskikh, Eduard Gorbunov, Alexander Gasnikov, Pavel Dvurechensky, César A. Uribe

***Abstract*— We introduce primal and dual stochastic gradient oracle methods for distributed convex optimization problems over networks. We show that the proposed methods are optimal (in terms of communication steps) for primal and dual oracles. Additionally, for a dual stochastic oracle, we propose a new analysis method for the rate of convergence in terms of duality gap and probability of large deviations. This analysis is based on a new technique that allows to bound the distance between the iteration sequence and the optimal point. By the proper choice of batch size, we can guarantee that this distance equals (up to a constant) to the distance between the starting point and the solution.**

I. Introduction

Distributed algorithms have been prevalent in the control theory and machine learning communities since early 70s and 80s [1]–[3]. The structural flexibilities introduced by a networked structure has been particularly relevant for recent applications, such as robotics and resource allocation [4]–[8], where large quantities of data are involved, and generation and processing of information is not centralized [9]–[13].

A distributed system is usually modeled as a network of computing agents connected in a definite way. These agents can act as local processors or sensors, and have communication capabilities to exchange information with each other. Precisely, the communication between agents is subject to the constraints imposed by the network structure. The object of study of distributed optimization is then to design algorithms that can be locally executed by the agents, and that exploit the network communications to solve a network-wide global problem cooperatively [14], [15].

Formally, we consider the optimization problem of minimizing the finite sum of m convex functions

$$\min_{x\in\mathbb{R}^n} f(x) := \sum_{i=1}^{m} f_i(x), \tag{1}$$

The work of D. Dvinskikh and P. Dvurechensky in part III-B was funded by Russian Science Foundation (project 18-71-10108). The work of E. Gorbunov in part III-A was supported by RFBR 18-31-20005 mol-a-ved. The work of A. Gasnikov in part II was supported by RFBR 18-29-03071 mk.

D.D and P.D. are with the Weierstrass Institute for Applied Analysis and Stochastics, Germany, and the Institute for Information Transmission Problems, Russia ({*darina.dvinskikh,pavel.dvurechensky*}*@wias-berlin.de*). E.G. is with the Moscow Institute of Physics and Technology, Russia (*eduard.gorbunov@phystech.edu*). A.G. is with Moscow Institute of Physics and Technology, Institute for Information Transmission Problems, Russia and National Research University Higher School of Economics, Russia (*gasnikov@yandex.ru*). C.A.U. is with the the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, USA (*cauribe@mit.edu*).

where each agent $i = \{1, 2, \dots, m\}$ in the network has access to the function f_i only, and yet, we seek that every agent cooperatively achieves a solution of (1).

In this paper, we consider the stochastic version of problem (1), when $f_i(x) = \mathbb{E}\tilde{f}_i(x, \xi)$, and ξ is a random variable. We provide an accelerated primal gradient method and an accelerated dual gradient method for this stochastic problem and estimate, for each case, the number of communication steps in the network and the number of stochastic oracle calls in order to obtain a solution with high probability. We also discuss extensions of our algorithms under additional strong convexity assumption.

Optimal methods for distributed optimization over networks were recently proposed and analyzed [16], [17]. However, there were only studied for deterministic settings. In [18], the authors studied a primal-dual method for stochastic problems. The setting of the latter paper is close to what we consider as the primal approach, but our algorithm and analysis are different, and, unlike [18], we consider smooth primal problem. Other approaches for distributed stochastic optimization has been studied in the literature [19], [20]. In contrast, we provide optimal communication complexities, as well as explicit dependency on the network topology.

This paper is organized as follows. Section II describes a primal approach for the solution of stochastic distributed optimization problems. Section III presents our main result on the communication and oracle complexity of a dual based stochastic distributed optimization problems. Finally, in Section IV, conclusions and future work are presented.

Notation: We define the maximum eigenvalue and minimal non-zero eigenvalue of a symmetric matrix W as $\lambda_{\max}(W)$ and $\lambda^+_{\min}(W)$ respectively, and define the condition number of matrix W as $\chi(W)$. We denote by $\mathbf{1}_m$ the vector of ones in $\mathbb{R}^m$. Denoting by $\|\cdot\|_2$ the standard Euclidean norm, we say that a function f is M-Lipschitz if $\|\nabla f(x)\|_2 \leq M$, a function f is L-smooth if $\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x-y\|_2$, a function f is μ-strongly convex (μ-s.c.) if, for all $x, y \in \mathbb{R}^n$, $f(y) \geq f(x) + \langle \nabla f(x), y-x\rangle + \frac{\mu}{2}\|x-y\|_2^2$. Given $\beta \in (0,1)$, we denote $\rho_\beta = 1 + \ln(1/\beta) + \sqrt{\ln(1/\beta)}$.

Due to space limitations, we omit the proofs of the main lemmas and theorems. For a complete version see [21].

II. Primal distributed approaches

In this section, we study the problem of distributed stochastic optimization over networks from a primal approach. Initially, we analyze the optimal convergence rate of the consensus algorithm from the optimization point of view. This allows us to establish the number of communication

rounds necessary for a group of agents over a network to reach some approximate agreement on a consensus value. This communication complexity is explicitly stated in terms of the condition number of the graph Laplacian. Then, we propose a primal-based method that uses the consensus algorithm to reach some approximate agreement on the gradient values. We analyze the effects of the deviations from the consensus value at each of the agents as a form of inexactness in the gradient oracle.

A. Consensus Problem

Consider a network of m agents whose interactions are represented by a connected and undirected graph $G = (V, E)$ with the set V of m vertices and the set of edges $E = \{(i, j) : i, j \in V\}$. Thus, agent i can communicate with agent j if and only if $(i, j) \in E$. Assume that each agent i has its own vector vector $y_i^0 \in R^n$, and its goal is to find an approximation to the vector $y^* = \frac{1}{m}\sum_{i=1}^m y_i^0$ by performing communications with neighboring agents. To do this, consider the Laplacian of the graph G, to be defined as a matrix $\bar{W}$ with entries,

$$[\bar{W}]_{ij} = \begin{cases} -1, & \text{if } (i,j) \in E, \\ \deg(i), & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases}$$

where $\deg(i)$ is the degree of vertex i (i.e., the number of neighboring nodes). Let us denote $W = \bar{W} \otimes I_n$, where $\otimes$ denotes Kronecker product and I_n is the unit matrix. Now, consider the optimization problem

$$\min_{y \in \mathbb{R}^{mn}} g(y) := \frac{1}{2}\langle y, Wy\rangle. \tag{2}$$

Since the gradient of the objective in (2) coincides with the sparcity pattern of of W, which is orthogonal to the null space of W, gradient type methods generate a sequence belonging to this space: $\{y^k\}_{k=1}^N \in y^0 + \mathrm{Ker(W)}^\perp$. In the space $\mathrm{Ker(W)}^\perp$, the objective $g(y)$ is $\lambda_{\max}(W)$-smooth and $\lambda_{\min}^+(W)$-strongly convex.

Therefore, any vector y with equal components is a solution of (2). Note that $y_1 = \cdots = y_m \iff Wy = 0$ if and only if W is the Laplacian of a connected graph. Starting from an arbitrary point $y^0 = [y_1^0, \ldots, y_m^0]$, classical gradient methods [22], [23] converge to the exact solution $y^* = \frac{1}{m}\sum_{i=1}^m y_i^0 \cdot \mathbf{1}$ which has the natural interpretation that agents reached agreement on a common decision by exchanging the information with their immediate neighbors.

Solving problem (2) over a network of agents supposes that each agent i has access only to y_i. However, the sparcity structure of W, which is induced by the network topology, allows us to write a version of Nesterov's accelerated gradient method [22], [23], that can be excecuted in a distributed manner for $L = \lambda_{\max}(W)$ and $\mu = \lambda_{\min}^+(W)$, see Algorithm 1.

The next theorem, provides the iteration complexity of Algorithm 1. It shows the number of iterations, or communication rounds, required to reach a Δ-relative precision of solution for Problem 2.

Algorithm 1 Consensus Algorithm

Input: Starting point $y^0 = (y_1^0, \ldots, y_m^0)$, number of iterations N, $\varkappa(W) = \frac{\sqrt{\lambda_{\max}(W)} - \sqrt{\lambda_{\min}^+(W)}}{\sqrt{\lambda_{\max}(W)} + \sqrt{\lambda_{\min}^+(W)}}$

1: Each agent i do
2: $y_i^1 = y_i^0 - \frac{1}{\lambda_{\max}(W)}\sum_{j=1}^m W_{ij} y_j^0$
3: **for** $k = 1, \ldots N-1$ **do**
4:
$$y_i^{k+1} = y_i^k - \frac{1}{\lambda_{\max}(W)}\sum_{j=1}^m W_{ij}(y_j^k + \\ + \varkappa(W)(y_j^k - y_j^{k-1})) + \varkappa(W)(y_i^k - y_i^{k-1})$$

Output: $y_1^N, \ldots, y_m^N$

Theorem 1: Let $\Delta > 0$ some required relative precision, $N = O\left(\sqrt{\chi(W)}\ln\Delta\right)$, y^0 be some arbitrary initial point, and $y^* = \frac{1}{m}\sum_{i=1}^m y_i^0 \cdot \mathbf{1}$ be a solution for (2). Then, the output of Algorithm 1 has the following property:

$$\|y^N - y^*\|_2 \le \Delta\|y^0 - y^*\|_2. \tag{3}$$

Theorem 1 states the minimum number of communication rounds needed for Algorithm 1 to reach some $\Delta > 0$ accuracy on a solution for (2)

Next, we analyse how to use Algorithm 1 in the context of stochastic distributed optimization.

B. Finite Sum Minimization

Consider Problem (1), in the stochastic setting when $f_i(x) = \mathbb{E}\tilde{f}_i(x, \xi)$, for $i = 1, ..., m$, and ξ being a random variable. Now, we use the idea of consensus for designing a distributed accelerated gradient method for this problem. We assign each function f_i to the agent i and assume that this agent is able to calculate the stochastic gradient of f_i using a batch of size r, i.e., $\nabla^r \tilde{f}_i(x, \xi_i) = (1/r)\sum_{j=1}^r \nabla \tilde{f}_i(x, \xi_i^j)$. We assume that the network of agents is represented by the graph G as in the previous subsection.

At this point we use Algorithm 1 taking $y_i = m\nabla^r \tilde{f}_i(x, \xi_i)$ and N specified in Theorem 1 to find a δ-approximation to the stochastic gradient of f in (1), i.e.,

$$\widetilde{\nabla} f_i(x_i) = \sum_{i=1}^m \nabla^r \tilde{f}_i(x_i, \xi_i) + \delta \quad \forall i \in V. \tag{4}$$

Algorithm 2 describes a method, to distributedly solve problem (1). In particular, we assume that at each iteration, the set of agents, can run Algorithm 1 on their local gradients, and compute an inexact gradient. Contrary to existing literature on distributed optimization, we use the inexact gradient formulation [24]–[27] to analyze the communication and oracle complexity of this method.

Theorem 2 describes the iteration and communication complexity of Algorithm 2. It shows how many communication rounds, how many oracle calls and the size of the mini-batch required to guarantee that the output of Algorithm 2 is approximated with arbitrary precision.

Assumption 1: Each function f_i has the following properties:

Algorithm 2 Distributed Primal Algorithm

Input: Starting point $x^0 = z^0$, N, $\alpha_0 = 0$, r.

1: Each agent i do
2: **for** $k = 1, \ldots, N-1$ **do**
3: $A_{k+1} = A_k + \alpha_{k+1} = 2L\alpha_{k+1}^2$
4: $y_i^{k+1} = (A_k x_i^k + \alpha_{k+1} z_i^k)/A_{k+1}$
5: Compute $\widetilde{\nabla} f_i(x_i^{k+1})$, as in (4) with r, using Algorithm 1
6: $z_i^{k+1} = z_i^k - \alpha \widetilde{\nabla} f_i(x_i^{k+1})$
7: $x_i^{k+1} = (A_k x^k + \alpha_{k+1} z_i^{k+1})/A_{k+1}$

Output: x_i^N for each agent $i = 1, \ldots, m$

- For $i \in V$, $\mathbb{E}_{\xi_i} \nabla \tilde{f}_i(x, \xi_i) = \nabla f_i(x)$, for all x, where $\{\xi_i\}_{i=1}^m$ are i.i.d.
- For $i \in V$, $\mathbb{E}_{\xi_i} \exp\left(\|\nabla \tilde{f}_i(x, \xi_i) - \nabla f_i(x)\|_2^2/\sigma^2\right) \leq \exp(1)$, for all x, where $\{\xi_i\}_{i=1}^m$ are i.i.d.

Theorem 2: Let Assumption 1 hold, $\beta \in (0,1)$ be confidence level, $\varepsilon > 0$ be the desired accuracy, and R such that $\|x^* - x^0\| \leq R$. Moreover, assume that $f(x)$ is L-smooth and $\forall x \ \|\nabla \tilde{f}_i(x, \xi)\|_2 \leq M \ i = 1, \ldots, m$. Let Algorithm 2 be run for $N = O\left(\sqrt{LR^2/\varepsilon}\right)$ iterations with $\Delta = O\left((1/mM)\sqrt{L\varepsilon/N}\right)$ in Algorithm 1 and $r = O\left(\max\left\{1, m^2\sigma^2\rho_\beta/(\varepsilon^2 N)\right\}\right)$, where ρ_β is defined in Sect. I. Then, with probability at least $1-\beta$, the output $x_i^N, i = 1, \ldots, m$ generated by Algorithm 2 satisfies

$$\sum_{i=1}^m f_i(x_i^N) - \sum_{i=1}^m f_i(x^*) \leq \varepsilon.$$

It remains to estimate the number of communications and stochastic oracle calls. Each iteration of the algorithm, by Theorem 1, requires $O\left(\sqrt{\chi(W)} \ln \Delta\right)$ communications, which gives the number of communications. At each iteration the stochastic oracle is called r times and the total number of calls is Nr, which gives the number of stochastic oracle calls.

Thus, Theorem 2 states that Algorithm 2 requires $\widetilde{O}\left(\sqrt{(LR^2/\varepsilon)\chi(W)}\right)$ communication rounds and

$$O\left(\max\left\{\sqrt{\frac{LR^2}{\varepsilon}}, \frac{m^2\sigma^2\rho_\beta}{\varepsilon^2}\right\}\right).$$

stochastic gradient oracle calls in order to reach some arbitrary accuracy ε.

The result of Theorem 2 can be generalized for the case of strongly convex objective f. Using the restart technique [23], [28], [29] for Algorithm 2, we can obtain stochastic oracle complexity

$$\widetilde{O}\left(\max\left\{\sqrt{\frac{L}{\mu}}, \frac{m^2\sigma^2\rho_\beta}{\mu\varepsilon}\right\}\right)$$

and communication complexity $\widetilde{O}\left(\sqrt{(L/\mu)\chi(W)}\right)$. The idea of the algorithm is that we run Algorithm 2 for $O\left(\sqrt{L/\mu}\right)$ iterations to make the objective residual twice smaller than on the previous restart. Then, the number of restarts is logarithmic in $1/\varepsilon$.

C. Finite sum minimization on a spanning tree graph

The communication complexity of the primal approach described in Subsection II-B can be improved by a transition to a centralized topology by constructing a spanning tree for the given network presented by a graph G with the additional assumption that one has access to a representation of this graph. In particular, we notice that this spanning tree computation can be done in a distributed manner as well [30], [31]. This additional pre-processing allows us to obtain, centralized network topology (master-slaves architecture) for arbitrary graph: we take the root of the spanning tree as the master node, and assume that all other nodes are slaves. We also assign each function f_i to each slaves, and then we can organize the distributed process as follows: each slave i calculates its gradient $\nabla f_i(x_i^k)$ and then sends it to the master node, which aggregates all gradients and computes the gradient step. Then it sends calculated value x^{k+1} back to the child nodes. This process presents forward and backward propagation of gradients and the argument via spanning tree with updating the argument x^k at each iteration k.

Denoting the diameter of graph G by d one can improve the number of communications rounds in Theorem 2 to $O\left(d\sqrt{LR^2/\varepsilon}\right)$. Note that for any graph $d \leq \sqrt{\chi(W)}$. For example, for a star graph $d = 2$ and $\sqrt{\chi(W)} \sim \sqrt{m}$. This approach allows to have a better dependency on the topology of the network. However, this approach might not be tractable for recent applications when the network topology changes which time [32], which might require the computation of the spanning tree at each iteration.

III. Dual distributed approaches

In this section, we follow [16], [17], [33], [34] and use primal-dual accelerated gradient methods [35]–[39], and use a dual formulation of the distributed optimization problem to design a class of optimal algorithms that can be executed over a network. We assume that the network of agents is represented by the graph G and matrix W as in the subsection II-A.

First, we present the dual formulation of the distributed optimization problem for the deterministic case, and then we develop our novel analysis for the case of stochastic dual oracles.

We assume that for all $i = 1, \ldots, m$ function f_i can be represented as the Fenchel-Legendre transform

$$f_i(x) = \max_{y \in \mathbb{R}^n} \{\langle y, x \rangle - \varphi_i(y)\}.$$

Thus, we rewrite the problem (1) as follows

$$\max_{\substack{x_1, \ldots, x_m \in \mathbb{R}^n, \\ x_1 = \cdots = x_m}} -F(\mathbf{x}) := -\sum_{i=1}^m f_i(x_i) = \max_{\substack{x_1, \ldots, x_m \in \mathbb{R}^n, \\ \sqrt{W}\mathbf{x} = 0}} -\sum_{i=1}^m f_i(x_i), \quad (5)$$

where $\mathbf{x} = [x_1, \ldots, x_m]^T \in \mathbb{R}^{nm}$ is the stacked column vector.

Algorithm 3 Distributed Dual Algorithm

Input: Starting point $\bar{\boldsymbol{\lambda}}^0 = \bar{\mathbf{y}}^0 = \bar{\boldsymbol{\zeta}}^0 = \mathbf{x}^0 = 0$, number of iterations N, $C_0 = \alpha_0 = 0$.

1: Each agent i do
2: **for** $k = 0, \ldots, N-1$ **do**
3: $\alpha_{k+1} = \frac{k+2}{4L}$, $A_{k+1} = \sum_{i=1}^{k+1} \alpha_i$
4: $\bar{\lambda}_i^{k+1} = (\alpha_{k+1}\bar{\zeta}_i^k + A_k \bar{y}_i^k)/A_{k+1}$.
5: $\bar{\zeta}_i^{k+1} = \bar{\zeta}_i^k - \alpha_{k+1} \sum_{j=1}^m W_{ij} x_j(\bar{\lambda}_j^t)$.
6: $\bar{y}_i^{k+1} = (\alpha_{k+1}\bar{\zeta}_i^{k+1} + A_k \bar{y}_i^k)/A_{k+1}$.
7: $x_i^N = \frac{1}{A_N} \sum_{k=0}^N \alpha_k x_i(\bar{\lambda}_i^k)$.

Output: $\mathbf{x}^N$, $\bar{\mathbf{y}}^N$.

Then, we introduce the Lagrangian dual problem to problem (5) with dual variables $\mathbf{y} = [y_1^T, \cdots, y_m^T]^T \in \mathbb{R}^{mn}$ as

$$\min_{\mathbf{y}\in\mathbb{R}^{mn}} \max_{\mathbf{x}\in\mathbb{R}^{nm}} \sum_{i=1}^m \Big(\langle y_i, [\sqrt{W}\mathbf{x}]_i\rangle - f_i(x_i)\Big)$$
$$= \min_{\mathbf{y}\in\mathbb{R}^{mn}} \psi(\mathbf{y}) := \varphi(\sqrt{W}\mathbf{y}) := \sum_{i=1}^m \varphi_i([\sqrt{W}\mathbf{y}]_i), \quad (6)$$

where we used the notations $[\sqrt{W}\mathbf{x}]_i$ and $[\sqrt{W}\mathbf{y}]_i$ for describing the i-th n-dimensional block of vectors $\sqrt{W}\mathbf{x}$ and $\sqrt{W}\mathbf{y}$ respectively, and also we used the equality $\sum_{i=1}^m \langle \mathbf{y}_i, [\sqrt{W}\mathbf{x}]_i\rangle = \sum_{i=1}^m \langle [\sqrt{W}\mathbf{y}]_i, \mathbf{x}_i\rangle$.

Note that dealing with the dual problem does not oblige us to use dual oracle of $\nabla\varphi_i$. Indeed,

$$\nabla\varphi([\sqrt{W}\mathbf{y}]_i) = [\sqrt{W}\mathbf{x}(\sqrt{W}\mathbf{y})]_i, \quad (7)$$

where $x_i([W\mathbf{y}]_i) = \arg\max_{x_i\in\mathbb{R}^n} \Big\{\langle [\sqrt{W}\mathbf{x}]_i, y_i\rangle - f_i(x_i)\Big\}$. So we can use the primal oracle ∇f_i to solve this auxiliary subproblem and find an approximation to $\nabla\varphi_i$.

Making the change of variables $\bar{\mathbf{y}} := \sqrt{W}\mathbf{y}$ and structure of Laplacian matrix W allows us to present accelerated gradient method in a distributed manner for the dual problem.

Theorem 3: Let $\varepsilon > 0$ be a desired accuracy and assume that $\|\nabla F(\mathbf{x}^*)\|_2 = M_F$ and that the primal objective in (5) is μ-strongly convex. Then the sequences $\mathbf{x}^N$ and $\mathbf{y}^N$ generated by Algorithm 3 after $N = O\big(\sqrt{(M_F^2/\mu\varepsilon)\chi(W)}\big)$ iterations and oracle calls of dual function $\nabla\varphi_i$ per node $i = 1, \ldots m$ satisfy the following condition $F(\mathbf{x}^N) + \psi(\bar{\mathbf{y}}^N) \le \varepsilon$

Next, we focus on the case where we only have access to the stochastic dual oracle.

A. Dual Approach with Stochastic Dual Oracle

Now, we suppose that $\psi(\mathbf{y})$ is endowed with stochastic oracle $\nabla\psi(\mathbf{y}, \xi)$, satisfying the following conditions[1]:

$$\mathbb{E}_\xi \nabla\psi(\mathbf{y}, \xi) = \nabla\psi(\mathbf{y}) \quad \text{and}$$
$$\mathbb{E}_\xi \exp\big(\|\nabla\psi(\mathbf{y}, \xi) - \nabla\psi(\mathbf{y})\|_2^2/\sigma_\psi^2\big) \le \exp(1).$$

We assume that the function ψ is L_ψ-smooth. If, the primal objective is μ-strongly convex, then $L_\psi \le \lambda_{\max}(W)/\mu$.

[1]We believe that the light-tail assumption can be relaxed to a more general setting [40].

Algorithm 4 Dual Stochastic Algorithm

Input: Starting point $\boldsymbol{\lambda}^0 = \mathbf{y}^0 = \boldsymbol{\zeta}^0 = \mathbf{x}^0 = 0$, number of iterations N, $C_0 = \alpha_0 = 0$,

1: **for** $k = 0, \ldots, N-1$ **do**
2: $$A_{k+1} = A_k + \alpha_{k+1} = 2L_\psi \alpha_{k+1}^2 \quad (9)$$
3: $$\boldsymbol{\lambda}^{k+1} = (\alpha_{k+1}\boldsymbol{\zeta}^k + A_k \mathbf{y}^k)/A_{k+1}. \quad (10)$$
4: Calculate $\nabla^{r_{k+1}}\psi(\boldsymbol{\lambda}_{k+1}, \{\xi_s\}_{s=1}^{r_{k+1}})$ according to (8) with batch size
$$r_{k+1} = O\left(\max\left\{1, \sigma_\psi^2 \alpha_{k+1} \ln(N/\delta)/\varepsilon\right\}\right)$$
5: $$\boldsymbol{\zeta}^{k+1} = \boldsymbol{\zeta}^k - \alpha_{k+1}\nabla^{r_{k+1}}\psi(\boldsymbol{\lambda}_{k+1}, \{\xi_s\}_{s=1}^{r_{k+1}}). \quad (11)$$
6: $$\mathbf{y}^{k+1} = (\alpha_{k+1}\boldsymbol{\zeta}^{k+1} + A_k \mathbf{y}^k)/A_{k+1}. \quad (12)$$
7: Set $\mathbf{x}^N = \frac{1}{A_N}\sum_{k=0}^N \alpha_k \mathbf{x}(\boldsymbol{\lambda}^k, \{\xi_i\}_{i=1}^{r_k})$.

Output: $\mathbf{x}^N$, $\mathbf{y}^N$.

Moreover, we assume that we can construct an approximation for $\nabla\psi(\mathbf{y})$ using batches of size r in the following form:

$$\nabla^r\psi(\mathbf{y}, \{\xi_i\}_{i=1}^r) = \frac{1}{r}\sum_{i=1}^r \nabla\psi(\mathbf{y}, \xi_i). \quad (8)$$

Theorem 4: Assume that F is μ-strongly convex and $\|\nabla F(\mathbf{x}^*)\|_2 = M_F$. Let $\varepsilon > 0$ be a desired accuracy. Assume that at each iteration of Algorithm 4 the approximation for $\nabla\psi(\mathbf{y})$ is chosen according to (8) with batch size $r_k = O\big(\max\{1, \sigma_\psi^2\alpha_k \ln(N/\delta)/\varepsilon\}\big)$. Then, after $N = O\big(\sqrt{(M_F^2/\mu\varepsilon)\chi(W)}\big)$ iterations, the outputs $\mathbf{x}^N$ and $\mathbf{y}^N$ of Algorithm 4 satisfy

$$F(\mathbf{x}^N) - F(\mathbf{x}^*) \le \varepsilon, \quad \|\sqrt{W}\mathbf{x}^N\|_2 \le \varepsilon/R_{\mathbf{y}} \quad (13)$$

with probability at least $1 - 3\delta$, where $\delta \in (0, 1/3)$, $\ln(N/\delta) \ge 3$ and $R_{\mathbf{y}}$ is such that $\|\mathbf{y}^*\|_2 \le R_{\mathbf{y}}$, $\mathbf{y}^*$ being an optimal solution of the dual problem.

Moreover, the number of stochastic oracle calls for the dual function $\nabla\varphi_i$ per node $i = 1, \ldots m$ is

$$O\left(\max\left\{\frac{\sigma_\psi^2 M_F^2}{\varepsilon^2 \lambda_{\min}^+(W)} \ln\left(\frac{1}{\delta}\sqrt{\frac{M_F^2}{\mu\varepsilon}\chi(W)}\right), \sqrt{\frac{M_F^2}{\mu\varepsilon}\chi(W)}\right\}\right)$$

To prove the theorem we first state a number of technical lemmas.

Lemma 5: For the sequence α_{k+1} defined in (9) we have for all $k \ge 0$

$$\alpha_{k+1} \le \widetilde{\alpha}_{k+1} \stackrel{\text{def}}{=} \frac{k+2}{2L_\psi}. \quad (14)$$

Lemma 6: Let A, B, and $\{r_i\}_{i=0}^N$ be non-negative numbers such that for all $l = 1, \ldots, N$

$$\frac{1}{2}r_l^2 \le Ar_0^2 + B\frac{r_0}{N}\sqrt{\sum_{k=0}^{l-1}(k+2)r_k^2}. \quad (15)$$

Then $r_l \leq Cr_0$, where C is such positive number that $C^2 \geq \max\{1, 2A + 2BC\}$.
The proof of the Lemma is followed from induction.

Lemma 7: Let the sequences of non-negative numbers $\{\alpha_k\}_{k\geq 0}$, random non-negative variables $\{R_k\}_{k\geq 0}$ and random vectors $\{\eta^k\}_{k\geq 0}$ and $\{a^k\}_{k\geq 0}$ for all $l = 1, \ldots, N$ satisfy

$$\frac{1}{2}R_l^2 \leq A + u\sum_{k=0}^{l-1}\alpha_{k+1}\langle \eta^{k+1}, a^k\rangle + c\sum_{k=0}^{l-1}\alpha_{k+1}^2\|\eta^{k+1}\|_2^2 \quad (16)$$

where A is deterministic non-negative number, $\|a^k\|_2 \leq d\widetilde{R}_k$, $d \geq 1$ is some positive deterministic constant and $\widetilde{R}_k = \max\{\widetilde{R}_{k-1}, R_k\}$ for all $k \geq 1$, $\widetilde{R}_0 = R_0$, $\widetilde{R}_k$ depends only on $\eta_0, \ldots, \eta^k$.
Moreover, assume, vector a^k is a function of $\eta^0, \ldots, \eta^{k-1}$ $\forall k \geq 1$, a^0 is a deterministic vector, and $\forall k \geq 0$,

$$\mathbb{E}\left[\eta^k \mid \{\eta^j\}_{j=0}^{k-1}\right] = 0,$$
$$\mathbb{E}\left[\exp\left(\|\eta^k\|_2^2\sigma_k^{-2}\right) \mid \{\eta^j\}_{j=0}^{k-1}\right] \leq \exp(1), \quad (17)$$

$\alpha_{k+1} \leq \widetilde{\alpha}_{k+1} = D(k+2)$, $\sigma_k^2 \leq (C\varepsilon)/(\widetilde{\alpha}_{k+1}\ln(N/\delta))$ for some $D, C > 0$, $\varepsilon > 0$. If additionally $\varepsilon \leq HR_0^2/N^2$, then with probability at least $1 - 2\delta$ the inequalities

$$\widetilde{R}_l \leq JR_0 \quad \text{and} \quad (18)$$
$$u\sum_{k=0}^{l-1}\alpha_{k+1}\langle \eta^{k+1}, a^k\rangle + c\sum_{k=0}^{l-1}\alpha_{k+1}^2\|\eta^{k+1}\|_2^2 \leq \left(24cCDH + udC_1\sqrt{CDHJg(N)}\right)R_0^2 \quad (19)$$

hold $\forall l = 1, \ldots, N$ simultaneously. Here C_1 is some positive constant, $g(N) = \left(\ln(N/\delta) + \ln\ln(B/b)\right)/\ln(N/\delta)$,

$$B = 2d^2CDHR_0^2\Big(2A + ud\widetilde{R}_0^2 + 12CD\varepsilon\left(2c + ud\right)N(N+3)\Big)(2ud)^N,$$

$b = \sigma_0^2\widetilde{\alpha}_1^2d^2\widetilde{R}_0^2$ and

$$J = \max\Big\{1, udC_1\sqrt{CDHg(N)} + \sqrt{u^2d^2C_1^2CDHg(N) + \tfrac{2A}{R_0^2} + 48cCDH}\Big\}.$$

B. Example: Computation of Wasserstein Barycenters

It may seem that the problem with dual stochastic oracle is artificial. Next, we present the regularized Wasserstein barycenter problem [41]–[44], which is a recent example of a function with stochastic dual oracle,

$$\min_{p\in S_n(1)}\sum_{i=1}^m \mathcal{W}_{\mu,q_i}(p), \quad (20)$$

where $\mathcal{W}_{\mu,q_i}(p) = \min\limits_{\substack{\pi\mathbf{1}=p, \pi^T\mathbf{1}=q \\ \pi\geq 0}} \{\langle C, \pi\rangle + \mu\langle \pi\ln\pi\rangle\}$.
Here C is a transportation cost matrix, p, q are elements of standard probability simplex, logarithm of a matrix is taken componentwise. Problem (20) is not easily tractable in the distributed setting since cost of approximating of the gradient of $\mathcal{W}_{\mu,q_i}(p)$ requires to solve a large-scale minimization problem. On the other hand, as it is shown in [41],

$$\mathcal{W}_{\mu,q_i}(p) = \max_{u\in\mathbb{R}^n}\left\{\langle u, p\rangle - \mathcal{W}_{q,\mu}^*(u)\right\}$$
$$\mathcal{W}_{q,\mu}^*(u) = \mu\sum_{j=1}^n q_j\ln\left(\frac{1}{q_j}\sum_{i=1}^n\exp\left(\frac{-C_{ij}+u_i}{\mu}\right)\right).$$

So, the conjugate function has an explicit expression and its gradient can be calculated explicitly. Moreover, as the conjugate function has the form of finite-sum, we can use randomization and take a component i with probability q_i. As a corollary of our general Theorem 4, we obtain

Corollary 8: Taking the batch size $r_k = O\big((\sigma_\psi^2\alpha_k\ln(N/\beta)/\varepsilon\mu)\big)$, where $\sigma_\psi^2 = m\lambda_{\max}(W)$ after $N = O\big(\sqrt{(M_F^2/\mu\varepsilon)\chi(W)}\big)$ iterations the following holds for the output $\mathtt{p}^N$ of Algorithm 4 with probability at least $1 - 3\delta$, where $\delta \in (1, 1/3)$ is such that $(1 + \sqrt{\ln(1/\delta)})/\sqrt{\ln(N/\delta)} \leq 2$.

$$\sum_{i=1}^m \mathcal{W}_{\mu,q_i}(\mathtt{p}_i^N) - \sum_{i=1}^m \mathcal{W}_{\mu,q_i}(p^*) \leq \varepsilon, \ \|\sqrt{W}\mathtt{p}^N\|_2 \leq \varepsilon/R_{\mathbf{y}}.$$

Moreover, the total complexity per node is

$$O\left(n\max\left\{\frac{mM_F^2}{\varepsilon^2}\chi\ln\left(\frac{1}{\delta}\sqrt{\frac{M_F^2}{\mu\varepsilon}\chi}\right), \sqrt{\frac{M_F^2}{\mu\varepsilon}\chi}\right\}\right),$$

where ${M_F}^2 = 2nm\|C\|_\infty^2$ [42] and $\chi = \chi(W)$.

IV. Conclusion

We consider primal and dual distributed accelerated gradient methods for stochastic finite-sum minimization. One of the key features of our analysis are large deviations bounds for the error of the algorithms. Moreover, we show that the proposed methods have optimal communication complexity, up to logarithmic factors. For each of the proposed methods we provide an explicit oracle and communication complexity analysis. We illustrate the dual approach by the Wasserstein barycenter problem. As a future work we consider extending these results for different classes of problems, i.e., non-smooth and/or also strongly convex problems.

Acknowledgements: We are grateful to A. Nemirovski for fruitful discussions.

References

[1] V. Borkar and P. P. Varaiya, "Asymptotic agreement in distributed estimation," *IEEE Transactions on Automatic Control*, vol. 27, no. 3, pp. 650–655, 1982.
[2] J. N. Tsitsiklis and M. Athans, "Convergence and asymptotic agreement in distributed decision problems," *IEEE Transactions on Automatic Control*, vol. 29, no. 1, pp. 42–50, 1984.
[3] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, no. 345, pp. 118–121, 1974.
[4] L. Xiao and S. Boyd, "Optimal scaling of a gradient method for distributed resource allocation," *Journal of Optimization Theory and Applications*, vol. 129, no. 3, pp. 469–488, 2006.
[5] M. Rabbat and R. Nowak, "Decentralized source localization and tracking wireless sensor networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, 2004, pp. 921–924.

[6] T. Kraska, A. Talwalkar, J. C. Duchi, R. Griffith, M. J. Franklin, and M. I. Jordan, "Mlbase: A distributed machine-learning system." in *CIDR*, vol. 1, 2013, pp. 2–1.

[7] A. Nedić, A. Olshevsky, and C. A. Uribe, "Distributed learning for cooperative inference," *arXiv preprint arXiv:1704.02718*, 2017.

[8] A. Ivanova, P. Dvurechensky, and A. Gasnikov, "Composite optimization for the resource allocation problem," *arXiv:1810.00595*, 2018.

[9] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[10] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[11] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems." in *Conf. on Language Resources and Evaluation (LREC'08)*, 2016, pp. 3243–3249.

[12] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM Journal on Optimization*, vol. 27, no. 4, pp. 2597–2633, 2017.

[13] A. Nedić, A. Olshevsky, and C. A. Uribe, "Fast convergence rates for distributed non-Bayesian learning," *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5538–5553, Nov 2017.

[14] A. Nedić, A. Olshevsky, A. Ozdaglar, and J. N. Tsitsiklis, "On distributed averaging algorithms and quantization effects," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2506–2517, 2009.

[15] 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.

[16] K. Scaman, F. Bach, S. Bubeck, Y. T. Lee, and L. Massoulié, "Optimal algorithms for smooth and strongly convex distributed optimization in networks," in *Proc. of the 34th International Conference on Machine Learning*, 2017, pp. 3027–3036.

[17] C. A. Uribe, S. Lee, A. Gasnikov, and A. Nedić, "A dual approach for optimal algorithms in distributed optimization over networks," *arXiv:1809.00710*, 2018.

[18] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Mathematical Programming*, pp. 1–48, 2017.

[19] D. Jakovetic, D. Bajovic, A. K. Sahu, and S. Kar, "Convergence rates for distributed stochastic optimization over random networks," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 4238–4245.

[20] W. Li, M. Assaad, and P. Duhamel, "Distributed stochastic optimization in networks with low informational exchange," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2017, pp. 1160–1167.

[21] D. Dvinskikh, E. Gorbunov, A. Gasnikov, P. Dvurechensky, and C. A. Uribe, "On dual approach for distributed stochastic convex optimization over networks," *arXiv:1903.09844*, 2019.

[22] Y. Nesterov, *Introductory Lectures on Convex Optimization: a basic course*. Kluwer Academic Publishers, Massachusetts, 2004.

[23] ——, "A method of solving a convex programming problem with convergence rate $o(1/k^2)$," *Soviet Mathematics Doklady*, vol. 27, no. 2, pp. 372–376, 1983.

[24] A. d'Aspremont, "Smooth optimization with approximate gradient," *SIAM J. on Optimization*, vol. 19, no. 3, pp. 1171–1183, 2008.

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

[26] F. S. Stonyakin, D. Dvinskikh, P. Dvurechensky, A. Kroshnin, O. Kuznetsova, A. Agafonov, A. Gasnikov, A. Tyurin, C. A. Uribe, D. Pasechnyuk, and S. Artamonov, "Gradient methods for problems with inexact model of the objective," in *Mathematical Optimization Theory and Operations Research*, M. Khachay, Y. Kochetov, and P. Pardalos, Eds. Cham: Springer International Publishing, 2019, pp. 97–114, arXiv:1902.09001.

[27] L. Bogolubsky, P. Dvurechensky, A. Gasnikov, G. Gusev, Y. Nesterov, A. M. Raigorodskii, A. Tikhonov, and M. Zhukovskii, "Learning supervised pagerank with gradient-based and gradient-free optimization methods," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 4914–4922, arXiv:1603.00717.

[28] S. Ghadimi and G. Lan, "Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: Shrinking procedures and optimal algorithms," *SIAM J. on Optimization*, vol. 23, no. 4, pp. 2061–2089, 2013.

[29] P. Dvurechensky and A. Gasnikov, "Stochastic intermediate gradient method for convex problems with stochastic inexact oracle," *J. of Opt. Theory and Applications*, vol. 171, no. 1, pp. 121–145, 2016.

[30] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Transactions on Programming Languages and systems (TOPLAS)*, vol. 5, no. 1, pp. 66–77, 1983.

[31] H. Abdel-Wahab, I. Stoica, F. Sultan, and K. Wilson, "A simple algorithm for computing minimum spanning trees in the internet," *Information sciences*, vol. 101, no. 1-2, pp. 47–69, 1997.

[32] A. Rogozin, C. A. Uribe, A. Gasnikov, N. Malkovsky, and A. Nedić, "Optimal distributed optimization on slowly time-varying graphs," *arXiv:1805.06045*, 2018.

[33] K. Scaman, F. Bach, S. Bubeck, L. Massoulié, and Y. T. Lee, "Optimal algorithms for non-smooth distributed optimization in networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 2745–2754.

[34] M. Maros and J. Jaldén, "PANDA: A Dual Linearly Converging Method for Distributed Optimization Over Time-Varying Undirected Graphs," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 6520–6525.

[35] P. Dvurechensky, A. Gasnikov, E. Gasnikova, S. Matsievsky, A. Rodomanov, and I. Usik, "Primal-dual method for searching equilibrium in hierarchical congestion population games," in *Supplementary Proceedings of the 9th International Conference on Discrete Optimization and Operations Research and Scientific School (DOOR 2016) Vladivostok, Russia, September 19 - 23, 2016*, 2016, pp. 584–595, arXiv:1606.08988.

[36] A. Chernov, P. Dvurechensky, and A. Gasnikov, "Fast primal-dual gradient method for strongly convex minimization problems with linear constraints," in *Discrete Optimization and Operations Research: 9th International Conference, DOOR 2016, Vladivostok, Russia, September 19-23, 2016, Proceedings*, Y. Kochetov, M. Khachay, V. Beresnev, E. Nurminski, and P. Pardalos, Eds. Springer International Publishing, 2016, pp. 391–403.

[37] A. S. Anikin, A. V. Gasnikov, P. E. Dvurechensky, A. I. Tyurin, and A. V. Chernov, "Dual approaches to the minimization of strongly convex functionals with a simple structure under affine constraints," *Computational Mathematics and Mathematical Physics*, vol. 57, no. 8, pp. 1262–1276, 2017.

[38] P. Dvurechensky, A. Gasnikov, and A. Kroshnin, "Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn's algorithm," in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy and A. Krause, Eds., vol. 80, 2018, pp. 1367–1376, arXiv:1802.04367.

[39] S. V. Guminov, Y. E. Nesterov, P. E. Dvurechensky, and A. V. Gasnikov, "Accelerated primal-dual gradient descent with linesearch for convex, nonconvex, and nonsmooth optimization problems," *Doklady Mathematics*, vol. 99, no. 2, pp. 125–128, 2019.

[40] P. E. Dvurechensky, A. V. Gasnikov, and A. A. Lagunovskaya, "Parallel algorithms and probability of large deviation for stochastic convex optimization problems," *Numerical Analysis and Applications*, vol. 11, no. 1, pp. 33–37, 2018, arXiv:1701.01830.

[41] M. Cuturi and G. Peyré, "A smoothed dual approach for variational wasserstein problems," *SIAM J. on Imaging Sciences*, vol. 9, no. 1, pp. 320–343, 2016.

[42] A. Kroshnin, N. Tupitsa, D. Dvinskikh, P. Dvurechensky, A. Gasnikov, and C. Uribe, "On the complexity of approximating Wasserstein barycenters," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97, 2019, pp. 3530–3540, arXiv:1901.08686.

[43] C. A. Uribe, D. Dvinskikh, P. Dvurechensky, A. Gasnikov, and A. Nedić, "Distributed Computation of Wasserstein Barycenters Over Networks," in *2018 IEEE Conference on Decision and Control (CDC)*, Dec 2018, pp. 6544–6549.

[44] P. Dvurechensky, D. Dvinskikh, A. Gasnikov, C. A. Uribe, and A. Nedić, "Decentralize and randomize: Faster algorithm for Wasserstein barycenters," in *Advances in Neural Information Processing Systems 31*, 2018, pp. 10 783–10 793, arXiv:1806.03915.