Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Derivative-Free Method For Composite Optimization With Applications To Decentralized Distributed Optimization

Aleksandr Beznosikov * **Eduard Gorbunov** ** **Alexander Gasnikov** ***

* *Moscow Institute of Physics and Technology, Russia*
Sirius University of Science and Technology, Russia
(e-mail: beznosikov.an@phystech.edu).
** *Moscow Institute of Physics Technology, Russia*
Sirius University of Science and Technology, Russia
Institute for Information Transmission Problems RAS, Russia
(e-mail: eduard.gorbunov@phystech.edu).
*** *Moscow Institute of Physics and Technology, Russia*
Sirius University of Science and Technology, Russia
Institute for Information Transmission Problems RAS, Russia,
Caucasus Mathematical Center, Adyghe State University, Russia
(e-mail: gasnikov@yandex.ru).

Abstract: In this paper, we propose a new method based on the Sliding Algorithm from Lan (2016, 2019) for the convex composite optimization problem that includes two terms: smooth one and non-smooth one. Our method uses the stochastic noised zeroth-order oracle for the non-smooth part and the first-order oracle for the smooth part and it is the first method in the literature that uses such a mixed oracle for the composite optimization. We prove the convergence rate for the new method that matches the corresponding rate for the first-order method up to a factor proportional to the dimension of the space or, in some cases, its squared logarithm. We apply this method for the decentralized distributed optimization and derive upper bounds for the number of communication rounds for this method that matches known lower bounds. Moreover, our bound for the number of zeroth-order oracle calls per node matches the similar state-of-the-art bound for the first-order decentralized distributed optimization up to to the factor proportional to the dimension of the space or, in some cases, its squared logarithm.

Keywords: gradient sliding, zeroth-order optimization, decentralized distributed optimization, composite optimization

1. INTRODUCTION

In this paper we consider finite-sum minimization problem

$$\min_{x \in X \subseteq \mathbb{R}^n} f(x) = \frac{1}{m} \sum_{i=1}^{m} f_i(x), \qquad (1)$$

where each f_i is convex and differentiable function and X is closed and convex. Such kind of problems are highly widespread in machine learning applications Shalev-Shwartz and Ben-David (2014), statistics Spokoiny et al. (2012) and control theory Rao (2009). In particular, we are interested in the case when functions f_i are stored on different devices which are connected in a network Lan et al. (2017); Scaman et al. (2017, 2018, 2019); Dvinskikh et al. (2019); Dvinskikh and Gasnikov (2019); Gorbunov et al. (2019); Uribe et al. (2020). This scenario often appears when the goal is to accelerate the training of big machine learning models or when the information that defines f_i is known only to the i-th *worker*.

In the centralized or parallel case, the iteration of the method can be described in the following way:

1) each worker in parallel performs computations of either gradients or stochastic gradients of f_i;
2) then workers send the results (not necessarily gradients that they just computed) to one predefined node called *master* node;
3) master node processes received information and broadcast new information to each worker that is needed to get new iterate.

However, such an approach has several problems, e.g. synchronization drawback or high requirements to the master node. There are a lot of works that cope with aforementioned drawbacks (see Stich (2018); Karimireddy et al. (2019); Alistarh et al. (2017); Wen et al. (2017)).

Another possible approach to deal with these drawbacks is to use decentralized architecture Bertsekas and Tsitsiklis (1989). Essentially it means that workers are able to communicate only with their neighbors and communications are simultaneous. Moreover, such an approach is more robust, e.g. it can be applied to time-varying (wireless) communication networks Rogozin and Gasnikov (2019).

1.1 Our contributions

We develop a new method called Zeroth-Order Sliding Algorithm (zoSA) for solving convex composite problem containing non-smooth part and L-smooth part which uses biased stochastic zeroth-order oracle for the non-smooth term and first-order oracle for the smooth component which is, to the best of our knowledge, the first method that uses zeroth-order and first-order oracles for composite optimization problem in such a way (see the details in Section 3). We prove the convergence result for the proposed method that matches known results for the number of first-oracle calls. Regarding the non-smooth component, we prove that the required number of zeroth-order oracle calls is typically n times or, in some cases, $\log n$ times larger then the corresponding bound obtained for the number of first-order oracle calls required for the non-smooth part which is natural for the derivative-free optimization (see Larson et al. (2019)).

Next, we show how to apply zoSA to the decentralized distributed optimization and get results that match the state-of-the-art results for the first-order non-smooth decentralized distributed optimization in terms of the communication rounds.

One can find the proofs together with the extension of zoSA to the case when the smooth part is additionally strongly convex and numerical experiments in the full version of this paper available on arXiv Beznosikov et al. (2019).

2. NOTATION AND DEFINITIONS

We use $\langle x, y\rangle \overset{\text{def}}{=} \sum_{i=1}^n x_i y_i$ to denote standard inner product of $x, y \in \mathbb{R}^n$ where x_i corresponds to the i-th component of x in the standard basis in $\mathbb{R}^n$. It induces ℓ_2-norm in $\mathbb{R}^n$ in the following way $\|x\|_2 \overset{\text{def}}{=} \sqrt{\langle x, x\rangle}$. We denote ℓ_p-norms as $\|x\|_p \overset{\text{def}}{=} \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$ for $p \in (1, \infty)$ and for $p = \infty$ we use $\|x\|_\infty \overset{\text{def}}{=} \max_{1\le i\le n} |x_i|$. The dual norm $\|\cdot\|_*$ for the norm $\|\cdot\|$ is defined in the following way: $\|y\|_* \overset{\text{def}}{=} \max\{\langle x, y\rangle \mid \|x\| \le 1\}$. To denote maximal and minimal positive eigenvalues of positive semidefinite matrix $A \in \mathbb{R}^{n\times n}$ we use $\lambda_{\max}(A)$ and $\lambda^+_{\min}(A)$ respectively and we use $\chi(A) \overset{\text{def}}{=} \lambda_{\max}(A)/\lambda^+_{\min}(A)$ to denote condition number of A. Operator $\mathbb{E}[\cdot]$ denotes full mathematical expectation and operator $\mathbb{E}_\xi[\cdot]$ express conditional mathematical expectation w.r.t. all randomness coming from random variable ξ. To define the Kronecker product of two matrices $A \in \mathbb{R}^{m\times m}$ and $B \in \mathbb{R}^{n\times n}$ we use $A \otimes B \in \mathbb{R}^{nm\times nm}$. The identity matrix of the size $n \times n$ is denoted in our paper by I_n.

Since all norms in finite dimensional space are equivalent, there exist such constants C_1, C_2 and C_3 that for all $x \in \mathbb{R}^n$

$$\|x\|_* \le C_1\|x\|_2, \quad \|x\|_2 \le C_2\|x\|_*, \quad \|x\| \le C_3\|x\|_2. \quad (2)$$

For example, if $\|\cdot\| = \|\cdot\|_2$, then $C_1 = C_2 = C_3 = 1$ and if $\|\cdot\| = \|\cdot\|_1$, then $\|\cdot\|_* = \|\cdot\|_\infty$ and $C_1 = 1$, $C_2 = C_3 = \sqrt{n}$.

Definition 1. (L-smoothness). Function g is called L-smooth in $X \subseteq \mathbb{R}^n$ with $L > 0$ w.r.t. norm $\|\cdot\|$ when it is differentiable and its gradient is L-Lipschitz continuous in X, i.e.

$$\|\nabla g(x) - \nabla g(y)\|_* \le L\|x - y\|, \quad \forall x, y \in X.$$

One can show that L-smoothness implies (see Nesterov (2004))

$$g(x) \le g(y) + \langle \nabla g(y), x - y\rangle + \frac{L}{2}\|x - y\|^2, \quad \forall x, y \in X. \quad (3)$$

Definition 2. (s-neighborhood of a set). For a given set $X \subseteq \mathbb{R}^n$ and $s > 0$ the s-neighborhood of X w.r.t. norm $\|\cdot\|$ is denoted by X_s which is defined as $X_s \overset{\text{def}}{=} \{z \in \mathbb{R}^n \mid \exists\, x \in X : \|y - x\| \le s\}$.

Definition 3. (Bregman divergence). Assume that function $\nu(x)$ is 1-strongly convex w.r.t. $\|\cdot\|$-norm and differentiable on X function. Then for any two points $x, y \in X$ we define Bregman divergence $V(x, y)$ associated with $\nu(x)$ as follows:

$$V(x, y) \overset{\text{def}}{=} \nu(y) - \nu(x) - \langle \nabla\nu(x), y - x\rangle.$$

Note that 1-strong convexity of $\nu(x)$ implies

$$V(x, y) \ge \frac{1}{2}\|x - y\|^2. \quad (4)$$

Finally, we denote the Bregman-diameter of the set X w.r.t. $V(x, y)$ as $D_{X,V} \overset{\text{def}}{=} \max\{\sqrt{2V(x, y)} \mid x, y \in X\}$. In view of (4) $D_{X,V}$ is an upper bound for the standard diameter of the set $D_X \overset{\text{def}}{=} \max\{\|x - y\| \mid x, y \in X\}$. When $V(x, y) = \frac{1}{2}\|x - y\|_2^2$ (standard Euclidean proximal setup) we have $D_{X,V} = D_X$. If $\|\cdot\| = \|\cdot\|_1$ is ℓ_1-norm, then in the case when X is a probability simplex, i.e. $X = \{x \in \mathbb{R}^n_+ \mid \sum_{i=1}^n x_i = 1\}$, and the distance generating function $\nu(x)$ is entropic, i.e. $\nu(x) = \sum_{i=1}^n x_i \ln x_i$, we have that $V(x, y)$ is the Kullback-Leibler divergence, i.e. $V(x, y) = \sum_{i=1}^n x_i \ln \frac{x_i}{y_i}$, and $D_{X,V} = \sqrt{2\ln n}$ (see Ben-Tal and Nemirovski (2015)).

3. MAIN RESULT

We consider the composite optimization problem

$$\min_{x\in X} \Psi_0(x) = f(x) + g(x), \quad (5)$$

where $X \subseteq \mathbb{R}^n$ is a compact and convex set with diameter D_X in $\|\cdot\|$-norm, function g is convex and L-smooth on X, f is convex differentiable function on X. Assume that we have an access to the first-order oracle for g, i.e. gradient $\nabla g(x)$ is available, and to the biased stochastic zeroth-order oracle for f (see also Gorbunov et al. (2018)) that for a given point x returns noisy value $\tilde{f}(x)$ such that

$$\tilde{f}(x) \overset{\text{def}}{=} f(x, \xi) + \Delta(x) \quad (6)$$

where $\Delta(x)$ is a bounded noise of unknown nature

$$|\Delta(x)| \le \Delta \quad (7)$$

and random variable ξ is such that

$$\mathbb{E}[f(x, \xi)] = f(x), \quad (8)$$

Additionally, we assume that for all $x \in X_s$ ($s \le D_X$)

$$\|\nabla f(x, \xi)\|_2 \le M(\xi), \quad \mathbb{E}[M^2(\xi)] = M^2. \quad (9)$$

This assumption implies that for all $x \in X_s$

$$|f(x, \xi) - f(y, \xi)| \le M(\xi)\|x - y\|_2, \quad \|\nabla f(x)\|_2 \le M.$$

Using this one can construct a stochastic approximation of $\nabla f(x)$ via finite differences (see Nesterov and Spokoiny (2017); Shamir (2017)):

$$\tilde{f}'_r(x) = \frac{n}{2r}(\tilde{f}(x+re) - \tilde{f}(x-re))e \tag{10}$$

where e is a random vector uniformly distributed on the Euclidean sphere and

$$r < sC_3 \tag{11}$$

is a smoothing parameter. Inequality (11) guarantees that the considered approximation requires points only from s-neighborhood of X since $\|re\| \leq rC_3$ (see (2)). Therefore, throughout the paper we assume that (11) holds. Following Shamir (2017) we assume that there exists such constant $p_* > 0$ that

$$\sqrt[4]{\mathbb{E}[\|e\|_*^4]} \leq p_*. \tag{12}$$

For example, when $\|\cdot\| = \|\cdot\|_2$ we have $p_* = 1$ and for the case when $\|\cdot\| = \|\cdot\|_1$ one can show that $p_* = O\left(\sqrt{\ln(n)/n}\right)$ (see Corollaries 2 and 3 from Shamir (2017)). Consider also the smoothed version

$$F(x) \stackrel{\text{def}}{=} \mathbb{E}_e[f(x+re)] \tag{13}$$

of $f(x)$ which is a differentiable in x function. In the following we summarize key properties of $F(x)$.

Lemma 1. (see also Lemma 8 from Shamir (2017)). Assume that differentiable function f defined on X_s satisfy $\|\nabla f(x)\|_2 \leq M$ with some constant $M > 0$. Then $F(x)$ defined in (13) is convex, differentiable and $F(x)$ satisfies

$$\sup_{x\in X} |F(x) - f(x)| \leq rM, \tag{14}$$

$$\nabla F(x) = \mathbb{E}_e\left[\frac{n}{r} f(x+re)e\right], \tag{15}$$

$$\|\nabla F(x)\|_* \leq \tilde{c} p_* \sqrt{n} M, \tag{16}$$

where $\tilde{c}$ is some positive constant independent of n and p_* is defined in (12).

In other words, $F(x)$ provides a good approximation of $f(x)$ for small enough r. Therefore, instead of solving (5) directly one can focus on the problem

$$\min_{x\in X} \Psi(x) \stackrel{\text{def}}{=} F(x) + g(x) \tag{17}$$

with small enough r since the difference between optimal values for (5) and (17) is at most rM. The following lemma establishes useful relations between $\nabla F(x)$ and $\tilde{f}'_r(x)$ defined in (10).

Lemma 2. (modification of Lemma 10 from Shamir (2017)). For $\tilde{f}'_r(x)$ defined in (10) the following inequalities hold:

$$|\mathbb{E}[\tilde{f}'_r(x)] - \nabla F(x)| \leq \frac{n\Delta}{r}, \tag{18}$$

$$\mathbb{E}[\|\tilde{f}'_r(x)\|_*^2] \leq 2p_*^2\left(cnM^2 + \frac{n^2\Delta^2}{r^2}\right), \tag{19}$$

where c is some positive constant independent of n.

In other words, one can consider $\tilde{f}'_r(x)$ as a biased stochastic gradient of $F(x)$ with bounded second moment and apply Stochastic Gradient Sliding from Lan (2016, 2019) with this stochastic gradient to solve problem (17).

Algorithm 1 Zeroth-Order Sliding Algorithm (`zoSA`)

Input: Initial point $x_0 \in X$ and iteration limit N.
Let $\beta_k \in \mathbb{R}_{++}, \gamma_k \in \mathbb{R}_+$, and $T_k \in \mathbb{N}$, $k = 1, 2, \ldots$, be given and set $\overline{x}_0 = x_0$.
for $k = 1, 2, \ldots, N$ **do**
1. Set $\underline{x}_k = (1-\gamma_k)\overline{x}_{k-1} + \gamma_k x_{k-1}$, and let $h_k(\cdot) \equiv l_g(\underline{x}_k, \cdot)$ be defined in (22).
2. Set
$$(x_k, \tilde{x}_k) = \mathtt{PS}(h_k, x_{k-1}, \beta_k, T_k);$$
3. Set $\overline{x}_k = (1-\gamma_k)\overline{x}_{k-1} + \gamma_k \tilde{x}_k$.

end for
Output: $\overline{x}_N$.

The `PS` (prox-sliding) procedure.
procedure: $(x^+, \tilde{x}^+) = \mathtt{PS}(h, x, \beta, T)$
Let the parameters $p_t \in \mathbb{R}_{++}$ and $\theta_t \in [0,1]$, $t = 1, \ldots$, be given. Set $u_0 = \tilde{u}_0 = x$.
for $t = 1, 2, \ldots, T$ **do**

$$u_t = \operatorname*{argmin}_{u\in X} \Big\{ h(u) + \langle \tilde{f}'_r(u_{t-1}), u\rangle + \beta V(x,u) + \beta p_t V(u_{t-1}, u) \Big\}, \tag{20}$$

$$\tilde{u}_t = (1-\theta_t)\tilde{u}_{t-1} + \theta_t u_t. \tag{21}$$

end for
Set $x^+ = u_T$ and $\tilde{x}^+ = \tilde{u}_T$.
end procedure:

In the Algorithm 1 we use the following function

$$l_g(x,y) \stackrel{\text{def}}{=} g(x) + \langle \nabla g(x), y - x\rangle. \tag{22}$$

At each iteration of `PS` subroutine the new direction e is sampled independently from previous iterations. We emphasize that we do not need to compute values of $F(x)$ which in the general case requires numerical computation of integrals over a sphere. In contrast, our method requires to know only noisy values of f defined in (6).

Next, we present the convergence analysis of `zoSA` that relies on the analysis for the Gradient Sliding method from Lan (2016, 2019). The following lemma provides an analysis of the subroutine `PS` from Algorithm 1.

Lemma 3. (see also Proposition 8.3 from Lan (2019)). Let $\{p_t\}_{t\geq 1}$ and $\{\theta_t\}_{t\geq 1}$ in the subroutine `PS` of Algorithm 1 satisfy

$$\theta_t = \frac{P_{t-1} - P_t}{(1-P_t)P_{t-1}}, \tag{23}$$

$$P_t = \begin{cases} 1 & t = 0, \\ p_t(1+p_t)^{-1}P_{t-1} & t \geq 1. \end{cases}$$

Then for any $t \geq 1$ and $u \in X$:

$$\begin{aligned} &\beta(1-P_t)^{-1}V(u_t,u) + [\Phi(\tilde{u}_t) - \Phi(u)] \\ &\quad \leq \beta P_t(1-P_t)^{-1}V(u_0,u) \\ &\quad + P_t(1-P_t)^{-1}\sum_{i=1}^{t}(p_i P_{i-1})^{-1}\Big[\frac{(\tilde{M} + \|\delta_i\|_*)^2}{2\beta p_i} \\ &\quad + \langle \delta_i, u - u_{i-1}\rangle\Big], \end{aligned} \tag{24}$$

where

$$\Phi(u) = h(u) + F(u) + \beta V(x,u), \tag{25}$$
$$\delta_t = \tilde{f}'_r(u_{t-1}) - \nabla F(u_{t-1}). \tag{26}$$

$$\tilde{M} = c\sqrt{n}C_1 M,$$

c is some positive constant independent of n, C_1 is from (2).

Using the lemma above we derive the main result.

Theorem 1. Assume that $\{p_t\}_{t\geq 1}$, $\{\theta_t\}_{t\geq 1}$, $\{\beta_k\}_{k\geq 1}$, $\{\gamma_k\}_{k\geq 1}$ in Algorithm 1 satisfy (23) and

$$\gamma_1 = 1, \quad \beta_k - L\gamma_k \geq 0, \quad k \geq 1, \tag{27}$$
$$\frac{\gamma_k\beta_k}{\Gamma_k(1-P_{T_k})} \leq \frac{\gamma_{k-1}\beta_{k-1}}{\Gamma_{k-1}(1-P_{T_{k-1}})}, \quad k \geq 2. \tag{28}$$

Then

$$\begin{aligned}&\mathbb{E}[\Psi(\overline{x}_N) - \Psi(x^*)] \\ &\leq \frac{\Gamma_N\beta_1}{1-P_{T_1}}V(x_0,u) + \Gamma_N\sum_{k=1}^{N}\sum_{i=1}^{T_k}\left[\frac{(\tilde{M}^2+\sigma^2)\gamma_k P_{T_k}}{\beta_k\Gamma_k(1-P_{T_k})p_i^2 P_{i-1}}\right. \\ &\qquad \left. + \frac{n\Delta D_X p_*}{r}\cdot\frac{\gamma_k P_{T_k}}{\Gamma_k(1-P_{T_k})p_i P_{i-1}}\right], \end{aligned} \tag{29}$$

where x^* is an arbitrary optimal point for (17), P_t is from (23),

$$\Gamma_k = \begin{cases}1, & k=1,\\ (1-\gamma_k)\Gamma_{k-1}, & k>1\end{cases} \tag{30}$$

and

$$\sigma^2 \overset{\text{def}}{=} 4p_*^2\left(CnM^2 + \frac{n^2\Delta^2}{r^2}\right), \tag{31}$$

where C is some positive constant independent of n.

The next corollary suggests the particular choice of parameters and states convergence guarantees in a more explicit way.

Corollary 1. Suppose that $\{p_t\}_{t\geq 1}$, $\{\theta_t\}_{t\geq 1}$ are

$$p_t = \frac{t}{2}, \quad \theta_t = \frac{2(t+1)}{t(t+3)}, \quad \forall t \geq 1, \tag{32}$$

N is given, $\{\beta_k\}$, $\{\gamma_k\}$, T_k are

$$\beta_k = \frac{2L}{k}, \quad \gamma_k = \frac{2}{k+1}, \quad T_k = \frac{N(\tilde{M}^2+\sigma^2)k^2}{\tilde{D}L^2} \tag{33}$$

for $\tilde{D} = {}^{3D_{X,V}^2}/{}_4$. Then $\forall N \geq 1$

$$\mathbb{E}[\Psi(\overline{x}_N) - \Psi(x^*)] \leq \frac{12LD_{X,V}^2}{N(N+1)} + \frac{n\Delta D_X p_*}{r}. \tag{34}$$

Finally, we extend the result above to the initial problem (5).

Corollary 2. Under the assumptions of Corollary 1 we have that the following inequality holds for all $N \geq 1$:

$$\begin{aligned}\mathbb{E}[\Psi_0(\overline{x}_N) - \Psi_0(x^*)] &\leq 2rM + \frac{12LD_{X,V}^2}{N(N+1)} \\ &\quad + \frac{n\Delta D_X p_*}{r}.\end{aligned} \tag{35}$$

From (35) it follows that if

$$r = \Theta\left(\frac{\varepsilon}{M}\right), \quad \Delta = O\left(\frac{\varepsilon^2}{nMD_X\min\{p_*,1\}}\right) \tag{36}$$

and $\varepsilon = O(\sqrt{n}MD_X)$, $s = \Omega({}^{\varepsilon}/{}_{MC_3})$, then the number of evaluations for ∇g and $\tilde{f}'_r$, respectively, required by Algorithm 1 to find an ε-solution of (5), i.e. such $\overline{x}_N$ that $\mathbb{E}[\Psi_0(\overline{x}_N)] - \Psi_0(x^*) \leq \varepsilon$, can be bounded by

$$O\left(\sqrt{\frac{LD_{X,V}^2}{\varepsilon}}\right), \tag{37}$$

$$O\left(\sqrt{\frac{LD_{X,V}^2}{\varepsilon}} + \frac{D_{X,V}^2 nM^2(C_1^2+p_*^2)}{\varepsilon^2}\right). \tag{38}$$

Let us discuss the obtained result and especially bounds (37) and (38). First of all, consider Euclidean proximal setup, i.e. $\|\cdot\| = \|\cdot\|_2$, $V(x,y) = \frac{1}{2}\|x-y\|_2^2$, $D_{X,V} = D_X$. In this case we have $p_* = C_1 = C_2 = C_3 = 1$ and bound (38) for the number of (6) oracle calls reduces to

$$O\left(\sqrt{\frac{LD_X^2}{\varepsilon}} + \frac{D_X^2 nM^2}{\varepsilon^2}\right)$$

and the number of $\nabla g(x)$ computations remains the same. It means that our result gives the same number of first-order oracle calls as in the original Gradient Sliding algorithm, while the number of the biased stochastic zeroth-order oracle calls is n times larger in the leading term than in the analogous bound from the original first-order method. In the Euclidean case our bounds reflect the classical dimension dependence for the derivative-free optimization (see Larson et al. (2019)).

Secondly, we consider the case when X is the probability simplex in $\mathbb{R}^n$ and the proximal setup is entropic (see the end of Section 2). As we mentioned earlier in Section 2 and in the beginning of this section, in this situation we have $D_{X,V} = \sqrt{2\ln n}$, $D_X = 2$, $p_* = O({}^{\ln(n)}/{}_n)$ and $C_1 = 1$, $C_2 = C_3 = \sqrt{n}$. Then number of $\nabla g(x)$ calculations is bounded by $O\left(\sqrt{{}^{(L\ln^2 n)}/{}_{\varepsilon}}\right)$. As for the number of $\tilde{f}'_r(x)$ computations, we get the following bound:

$$O\left(\sqrt{\frac{L\ln^2 n}{\varepsilon}} + \frac{M^2\ln^2 n}{\varepsilon^2}\right).$$

Clearly, in this case we have only polylogarithmical dependence on the dimension instead.

4. FROM COMPOSITE OPTIMIZATION TO CONVEX OPTIMIZATION WITH AFFINE CONSTRAINTS AND DECENTRALIZED DISTRIBUTED OPTIMIZATION

4.1 Convex Optimization with Affine Constraints

As an intermediate step between the composite optimization problem (5) and decentralized distributed optimization we consider the following problem

$$\min_{Ax=0, x\in X} f(x), \tag{39}$$

where $A \succeq 0$ and $\text{Ker}A \neq \{0\}$ and X is convex compact in $\mathbb{R}^n$ with diameter D_X. The dual problem for (39) can be written in the following way

$$\min_{y} \psi(y), \quad \text{where} \tag{40}$$
$$\begin{aligned}\varphi(y) &= \max_{x\in X}\left\{\langle y, x\rangle - f(x)\right\},\\ \psi(y) &= \varphi(A^\top y) = \max_{x\in Q}\left\{\langle y, Ax\rangle - f(x)\right\}\\ &= \langle y, Ax(A^\top y)\rangle - f(x(A^\top y))\\ &= \langle A^\top y, x(A^\top y)\rangle - f(x(A^\top y)),\end{aligned}$$

where $x(y) \stackrel{\text{def}}{=} \text{argmax}_{x\in X}\left\{\langle y, x\rangle - f(x)\right\}$. The solution of (40) with the smallest ℓ_2-norm is denoted in this paper as y_*. This norm $R_y \stackrel{\text{def}}{=} \|y_*\|_2$ can be bounded as follows Lan et al. (2017):

$$R_y^2 \le \frac{\|\nabla f(x^*)\|_2^2}{\lambda_{\min}^+(A^\top A)}.$$

Following Gasnikov (2018); Dvinskikh and Gasnikov (2019); Gorbunov et al. (2019) we consider the penalized problem

$$\min_{x\in X} F(x) = f(x) + \frac{R_y^2}{\varepsilon}\|Ax\|_2^2, \tag{41}$$

where $\varepsilon > 0$ is some positive number. It turns out (see the details in Gorbunov et al. (2019)) that if we have such $\hat{x}$ that $F(\hat{x}) - \min_{x\in X} F(x) \le \varepsilon$ then we also have

$$f(\hat{x}) - \min_{Ax=0, x\in X} f(x) \le \varepsilon, \quad \|A\hat{x}\|_2 \le \frac{2\varepsilon}{R_y}.$$

We notice that this result can be generalized in the following way: if we have such $\hat{x}$ that $\mathbb{E}[F(\hat{x})] - \min_{x\in X} F(x) \le \varepsilon$ then we also have

$$\mathbb{E}[f(\hat{x})] - \min_{Ax=0,x\in X} f(x) \le \varepsilon, \quad \sqrt{\mathbb{E}[\|A\hat{x}\|_2^2]} \le \frac{2\varepsilon}{R_y}. \tag{42}$$

Next, we consider the problem (41) as (5) with $g(x) = {}^{R_y^2\|Ax\|_2^2}/_{\varepsilon}$. Assume that $\|\nabla f(x)\|_2 \le M$ for all $x \in X$ and for f we have an access to the biased stochastic oracle defined in (6). We are interested in the situation when $\nabla g(x) = {}^{2R_y^2 A^\top Ax}/_{\varepsilon}$ can be computed exactly. Moreover, it is easy to see that $g(x)$ is ${}^{2R_y^2\lambda_{\max}(A^\top A)}/_{\varepsilon}$-smooth w.r.t. ℓ_2-norm. Applying Corollary 2 we get that in order to produce such a point $\hat{x}$ that satisfies (42) Algorithm 1 applied to solve (41) requires

$$O\left(\sqrt{\frac{\lambda_{\max}(A^\top A)R_y^2 D_X^2}{\varepsilon^2}}\right) \text{ calculations of } A^\top Ax$$

and

$$O\left(\sqrt{\frac{\lambda_{\max}(A^\top A)R_y^2 D_X^2}{\varepsilon^2}} + \frac{nD_X^2M^2}{\varepsilon^2}\right)$$

calculations of $\tilde{f}(x)$ since $p_* = C_2 = C_1 = 1$ for the Euclidean case. As we mentioned at the end of Section 3, this bound depends on dimension n in the classical way.

4.2 Decentralized Distributed Optimization

Now, we go back to the problem (1) and, following Scaman et al. (2017), we rewrite it in the distributed fashion:

$$\min_{\substack{x_1=\ldots=x_m\\ x_1,\ldots,x_m\in X}} f(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^m f_i(x_i), \tag{43}$$

where $\mathbf{x}^\top = (x_1^\top, \ldots, x_m^\top)^\top \in \mathbb{R}^{nm}$. Recall that we consider the situation when f_i is stored on the i-th node. In this case one can interpret x_i from (43) as a local variable of i-th node and $x_1 = \ldots = x_n$ as a consensus condition for the network. The common trick Scaman et al. (2017, 2018, 2019); Uribe et al. (2020) to handle this condition is to rewrite it using the notion of Laplacian matrix. In general, the Laplacian matrix $\overline{W} = \|\overline{W}_{ij}\|_{i,j=1,1}^{m,m} \in \mathbb{R}^{m\times m}$ of the graph G with vertices V, $|V| = m$ and edges V is defined as follows:

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

where $\deg(i)$ is degree of i-th node. In this paper we focus only on the connected networks. In this case $\overline{W}$ has unique eigenvector $\mathbf{1}_m \stackrel{\text{def}}{=} (1,\ldots,1)^\top \in \mathbb{R}^m$ associated to the eigenvalue 0. Using this one can show that for all vectors $a = (a_1, \ldots, a_m)^\top \in \mathbb{R}^m$ we have the following equivalence:

$$a_1 = \ldots = a_m \iff Wa = 0. \tag{44}$$

Using the Kronecker product $W \stackrel{\text{def}}{=} \overline{W} \otimes I_n$, which is also called Laplacian matrix for simplicity, one can generalize (44) for the n-dimensional case:

$$x_1 = \ldots = x_m \iff W\mathbf{x} = 0$$

and

$$x_1 = \ldots = x_m \iff \sqrt{W}\mathbf{x} = 0.$$

That is, instead of the problem (43) one can consider the equivalent problem

$$\min_{\substack{\sqrt{W}\mathbf{x}=0,\\ x_1,\ldots,x_m\in X}} f(\mathbf{x}) = \frac{1}{m}\sum_{i=1}^m f_i(x_i). \tag{45}$$

Next, we need to define parameters of f using local parameters of f_i. Assume that for each f_i we have $\|f_i(x_i)\|_2 \le M$ for all $x_i \in X$, all f_i are convex functions, the starting point is $\mathbf{x}_0^\top = (x_0^\top, \ldots, x_0^\top)^\top$ and $\mathbf{x}_*^\top = (x_*^\top, \ldots, x_*^\top)^\top$ is the optimality point for (45). Then, one can show (see Gorbunov et al. (2019) for the details) that $\|\nabla f(\mathbf{x})\|_2 \le {}^{M}/_{\sqrt{m}}$ on the set of such $\mathbf{x}$ that $x_1, \ldots, x_m \in X$, $D_{X^m}^2 = mD_X^2$ and $R_{\mathbf{y}}^2 \stackrel{\text{def}}{=} \|\mathbf{y}_*\|_2^2 \le {}^{M^2}/_{m\lambda_{\min}^+(W)}$.

Now we are prepared to apply results obtained in Section 4.1 to the problem (45). Indeed, this problem can be viewed as (45) with $A = \sqrt{W}$. Taking this into account, we conclude that one $A^\top Ax$ calculation corresponds to the calculation of Wx which can be computed during one communication round in the network with Laplacian matrix W. This simple observation implies that in order to produce such a point $\hat{\mathbf{x}}$ that satisfies (42) with $\hat{x} = \hat{\mathbf{x}}$, $A := \sqrt{W}$, $X := X^n$, $R_y := R_{\mathbf{y}}$ Algorithm 1 applied to the penalized problem (41) requires

$$O\left(\sqrt{\frac{\chi(W)M^2D_X^2}{\varepsilon^2}}\right) \text{ communication rounds}$$

and

$$O\left(\sqrt{\frac{\chi(W)M^2D_X^2}{\varepsilon^2}} + \frac{nD_X^2M^2}{\varepsilon^2}\right)$$

calculations of $\tilde{f}(x)$ per node since $p_* = 1$ for the Euclidean case. The bound for the communication rounds matches the lower bound from Scaman et al. (2018, 2019) and we conjecture that under our assumptions the obtained bound for zeroth-order oracle calculations per node is optimal up to polylogarithmic factors in the class of methods with optimal number of communication rounds (see also Dvinskikh and Gasnikov (2019); Gorbunov et al. (2019)).

5. DISCUSSION

To conclude, the proposed method — `zoSA` — is the first, to the best of our knowledge, $1/2$-order method for the convex composite optimization: it uses zeroth-order oracle for the non-smooth term and the first-order oracle for the smooth one. As for the future work, it would be interesting to study zeroth-order distributed methods for the smooth decentralized distributed optimization using the technique from Gorbunov et al. (2019). Another direction for future research is in developing the analysis of the proposed method for the case when X is unbounded and, in particular, when $X = \mathbb{R}^n$ via recurrences techniques from Gorbunov et al. (2018, 2019).

ACKNOWLEDGEMENTS

The research of A. Beznosikov, E. Gorbunov and A. Gasnikov was partially supported by RFBR, project number 19-31-51001. The research of E. Gorbunov was also partially supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye) 075-00337-20-03 and the research of A. Gasnikov was also partially supported by Yahoo! Research Faculty Engagement Program.

REFERENCES

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. (2017). QSGD: Communication-efficient SGD via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, 1709–1720.

Ben-Tal, A. and Nemirovski, A. (2015). *Lectures on Modern Convex Optimization (Lecture Notes)*. Personal web-page of A. Nemirovski.

Bertsekas, D.P. and Tsitsiklis, J.N. (1989). *Parallel and distributed computation: numerical methods*, volume 23. Prentice hall Englewood Cliffs, NJ.

Beznosikov, A., Gorbunov, E., and Gasnikov, A. (2019). Derivative-free method for decentralized distributed non-smooth optimization. *arXiv preprint arXiv:1911.10645*.

Dvinskikh, D., Gorbunov, E., Gasnikov, A., Dvurechensky, P., and Uribe, C.A. (2019). On primal and dual approaches for distributed stochastic convex optimization over networks. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, 7435–7440.

Dvinskikh, D. and Gasnikov, A. (2019). Decentralized and parallelized primal and dual accelerated methods for stochastic convex programming problems. *arXiv preprint arXiv:1904.09015*.

Gasnikov, A. (2018). Universal gradient descent. *MIPT*.

Gorbunov, E., Dvinskikh, D., and Gasnikov, A. (2019). Optimal decentralized distributed algorithms for stochastic convex optimization. *arXiv preprint arXiv:1911.07363*.

Gorbunov, E., Dvurechensky, P., and Gasnikov, A. (2018). An accelerated method for derivative-free smooth stochastic convex optimization. *arXiv preprint arXiv:1802.09022*.

Karimireddy, S.P., Rebjock, Q., Stich, S.U., and Jaggi, M. (2019). Error feedback fixes signsgd and other gradient compression schemes. *arXiv preprint arXiv:1901.09847*.

Lan, G. (2016). Gradient sliding for composite optimization. *Mathematical Programming*, 159(1-2), 201–235.

Lan, G. (2019). *Lectures on Optimization Methods for Machine Learning*. H. Milton Stewart School of Industrial and Systems Engineering Georgia Institute of Technology, Atlanta, GA.

Lan, G., Lee, S., and Zhou, Y. (2017). Communication-efficient algorithms for decentralized and stochastic optimization. *Mathematical Programming*, 1–48.

Larson, J., Menickelly, M., and Wild, S.M. (2019). Derivative-free optimization methods. *Acta Numerica*, 28, 287–404.

Nesterov, Y. (2004). *Introductory Lectures on Convex Optimization: a basic course*. Kluwer Academic Publishers, Massachusetts.

Nesterov, Y. and Spokoiny, V.G. (2017). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2), 527–566.

Rao, A.V. (2009). A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, 135(1), 497–528.

Rogozin, A. and Gasnikov, A. (2019). Projected gradient method for decentralized optimization over time-varying networks. *arXiv preprint arXiv:1911.08527*.

Scaman, K., Bach, F., Bubeck, S., Lee, Y.T., and Massoulié, L. (2017). Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 3027–3036. JMLR. org.

Scaman, K., Bach, F., Bubeck, S., Lee, Y.T., and Massoulié, L. (2019). Optimal convergence rates for convex distributed optimization in networks. *Journal of Machine Learning Research*, 20(159), 1–31.

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

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Shamir, O. (2017). An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52), 1–11.

Spokoiny, V. et al. (2012). Parametric estimation. finite sample theory. *The Annals of Statistics*, 40(6), 2877–2909.

Stich, S.U. (2018). Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*.

Uribe, C.A., Lee, S., Gasnikov, A., and Nedić, A. (2020). A dual approach for optimal algorithms in distributed optimization over networks. *Optimization Methods and Software*, 0(0), 1–40. doi:10.1080/10556788.2020.1750013. URL `https://doi.org/10.1080/10556788.2020.1750013`.

Wen, W., Xu, C., Yan, F., Wu, C., Wang, Y., Chen, Y., and Li, H. (2017). Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Advances in Neural Information Processing Systems*, 1509–1519.