ARTICLE TEMPLATE

Gradient-Type Methods For Decentralized Optimization Problems With Polyak-Łojasiewicz Condition Over Time-Varying Networks

Ilya Kuruzov[a] and Mohammad Alkousa[a,b] and Fedor Stonyakin[a,c] and Alexander Gasnikov[a,b,d]

[a]Moscow Institute of Physics and Technology, Dolgoprudny, Russia; [b]National Research University Higher School of Economics, Moscow, Russia; [c]V. Vernadsky Crimean Federal University, Simferopol, Republic of Crimea; [d]ISP RAS Research Center for Trusted Artificial Intelligence

ABSTRACT
This paper focuses on the decentralized optimization (minimization and saddle point) problems with objective functions that satisfy Polyak-Łojasiewicz condition (PL-condition). The first part of the paper is devoted to the minimization problem of the sum-type cost functions. In order to solve a such class of problems, we propose a gradient descent type method with a consensus projection procedure and the inexact gradient of the objectives. Next, in the second part, we study the saddle-point problem (SPP) with a structure of the sum, with objectives satisfying the two-sided PL-condition. To solve such SPP, we propose a generalization of the Multi-step Gradient Descent Ascent method with a consensus procedure, and inexact gradients of the objective function with respect to both variables. Finally, we present some of the numerical experiments, to show the efficiency of the proposed algorithm for the robust least squares problem.

1. Introduction

In this paper, firstly we study a sum-type minimization problem

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

where the functions f_i are generally non-convex and stored separately by nodes in a communication network, which is represented by a non-directed graph $\mathcal{G} = (E, V)$. The graph $\mathcal{G}$, possibly, can have a change over time structure. This problem, where the

object function depends on distributed data is typed as a decentralized optimization problem. We assume that f satisfies the well-known Polyak-Łojasiewicz condition (for brevity, we write PL-condition). This condition was originally introduced by Polyak [28], who proved that it is sufficient to show the global linear convergence rate for the gradient descent without assuming convexity. The PL-condition is very well studied by many researchers in many different works for many different settings of optimization problems and has been theoretically verified for objective functions of optimization problems arising in many practical problems. For example, it has been proven to be true for objectives of over-parameterized deep networks [6], learning LQR models [9], phase retrieval [37], Generative adversarial imitation learning of linear quadratic (see Example 3.1 in [3]). More discussions of PL-condition and many other simple problems can be found in [15].

This type of problems arises in different areas: distributed machine learning [17], resource allocation problem [13] and power system control [31].

The problem (1) can be reformulated as a problem with linear constraints. For this, let us assign each agent in the network a personal copy of parameter vector $x_i \in \mathbb{R}^d$ (column vector) and introduce

$$\mathbf{X} := \left(x_1^\top \ \ldots \ x_n^\top\right)^\top \in \mathbb{R}^{n \times d}, \quad F(\mathbf{X}) = \sum_{i=1}^n f_i(x_i). \tag{2}$$

Now we equivalently rewrite problem (1) as

$$\min_{\mathbf{X} \in \mathbb{R}^{n \times d}} F(\mathbf{X}) = \sum_{i=1}^n f_i(x_i), \quad \text{s.t. } x_1 = \cdots = x_n, \tag{3}$$

where it has the same optimal value as problem (1). This reformulation increases the number of variables but induces additional constraints at the same time.

Let us denote the set of consensus constraints $\mathcal{C} = \{\mathbf{X} | x_1 = \cdots = x_n\}$. Also, for each $\mathbf{X} \in \mathbb{R}^{n \times d}$ denote the average of its columns $\overline{x} = \frac{1}{n} \sum_{i=1}^n x_i \in \mathbb{R}^d$ and introduce its projection onto constraint set

$$\overline{\mathbf{X}} = \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top \mathbf{X} = \Pi_{\mathcal{C}}(\mathbf{X}) = \left(\overline{x}^\top \ \ldots \ \overline{x}^\top\right)^\top \in \mathbb{R}^{n \times d}.$$

Note that $\mathcal{C}$ is a linear subspace in $\mathbb{R}^{n \times d}$, and therefore projection operator $\Pi_{\mathcal{C}}$ is linear.

Secondly, we study the saddle-point problem with a structure of the sum

$$\min_{x \in \mathbb{R}^{d_x}} \max_{y \in \mathbb{R}^{d_y}} \left\{\phi(x, y) = \frac{1}{n} \sum_{i=1}^n \phi_i(x, y)\right\}, \tag{4}$$

where functions $\phi_i(\cdot, \cdot)$ are non-convex (with respect to the variable x for each y) and smooth (i.e., with Lipschitz-continuous gradient). These functions can be calculated only separately in different nodes in the communication network, which is represented by a non-directed graph $\mathcal{G}$.

Problems of type (4) arise in many applications, such that Generative adversarial network [10], adversarial training [18], and fair training [3]. In our work with problem

2

(4), we assume that $\phi(\cdot, y)$ and $-\phi(x, \cdot)$ satisfy the PL-condition (see Assumption (2.2), below).

Let $y(x) := \arg\max_{y \in \mathbb{R}^{d_y}} \phi(x, y)$, and let we set $f_i(x) := \phi_i(x, y(x))$, then problem (4) can be rewritten in the form of the minimization problem (1), i.e.,

$$\min_{x \in \mathbb{R}^{d_x}} \left\{ \max_{y \in \mathbb{R}^{d_y}} \left\{ \phi(x, y) = \frac{1}{n} \sum_{i=1}^{n} \phi_i(x, y) \right\} \right\} = \min_{x \in \mathbb{R}^{d_x}} \left\{ f(x) := \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\}. \qquad (5)$$

Let $\mathcal{C}_x := \left\{ \mathbf{X} \in \mathbb{R}^{n \times d_x} | x_1 = \ldots = x_n \right\}$, $\mathcal{C}_y := \left\{ \mathbf{Y} \in \mathbb{R}^{n \times d_y} | y_1 = \ldots = y_n \right\}$ and $\Phi(\mathbf{X}, \mathbf{Y}) := \sum_{i=1}^{n} \phi_i(x_i, y_i)$. So, we can rewrite (4), in the similar way like [32, 33], in the following form

$$\min_{\mathbf{X} \in \mathcal{C}_x} \max_{\mathbf{Y} \in \mathcal{C}_y} \Phi(\mathbf{X}, \mathbf{Y}). \qquad (6)$$

Similarly to the first proposed minimization problem (1), note that the problem (4), also can be rewritten in the same form

$$\min_{\mathbf{X} \in \mathcal{C}_x} F(\mathbf{X}) = \sum_{i=1}^{n} f_i(x_i). \qquad (7)$$

So, for solving the problem (5) (which is equivalent to problem (6)), we can try to use some gradient type algorithms, at each iteration of the used algorithm, we calculate the inexact gradient of $\phi(x, \cdot)$ for each $x \in \mathbb{R}^{d_x}$ in order to solve the (inner) maximization problem in (5).

In practice, Multi-step Gradient Descent Ascent (MGDA) algorithm and its modifications are widely used to solve the problem (4) (see e.g. [10, 11, 23]), as important algorithms for the considered type of saddle-point problems. In [25], authors demonstrate the effectiveness of MGDA on a class of games in which one of the players satisfies the PL-condition and another player has a general non-convex structure. At the same time, [41] shows that one step of gradient descent ascent demonstrates good performance and has theoretical guarantees for convergence in two-sided PL-games.

Finally, we mention that the main difference between distributed problems from usual optimization problems is to keep every agent's vectors to their average. It is approached through communication steps, where the communication can be performed in different scenarios. In our work, we will consider the time-varying network with changeable edges set and use the standard consensus procedure (see Subsec. 2.2) when the agent's vectors are averaged by multiplying by a weighted matrix of the graph at the current moment.

1.1. Related works

The decentralized algorithm makes two types of steps: local updates and information exchange. Local steps may use gradient [19, 26, 29, 30, 35, 42] or sub-gradient [27] computations. In primal-only methods, the agents compute gradients of their local functions and alternate taking gradient steps and communication procedures. Under cheap communication costs, it may be beneficial to replace a single consensus iteration with a series of information exchange rounds. Such methods as MSDA [34], D-NC [14],

and Mudag [42] employ multi-step gossip procedures. In order to achieve acceptable complexity bounds, one may distribute accelerated methods directly [7, 14, 19, 30, 42] or use a Catalyst framework [21]. Accelerated methods meet the lower complexity bounds for decentralized optimization [12, 22, 34]. As usual, by complexity we mean a sufficient number of iterations of the algorithm that guarantee the solution of the problem with a given accuracy.

Consensus restrictions $x_1 = \ldots = x_n$ may be treated as linear constraints, thus allowing for a dual reformulation of problem (1). Dual-based methods include dual ascent and its accelerated variants [34, 38, 40, 44]. Primal-dual approaches like ADMM [2, 39] are also implementable in decentralized scenarios. In [35], the authors developed algorithms for non-smooth convex objectives and provided lower complexity bounds for this case, as well.

Changing topology for time-varying networks requires new approaches to decentralized methods and a more complicated theoretical analysis. The first method with provable geometric convergence was proposed in [26]. Such primal algorithms as the Push-Pull Gradient Method [29] and DIGing [26] are robust to network changes and have theoretical guarantees of convergence over time-varying graphs. Recently, a dual method for time-varying architectures was introduced in [24].

In [33], it was studied the problem of decentralized optimization with strongly convex smooth objective functions. The authors investigated accelerated deterministic algorithms under time-varying network constraints with a consensus projection procedure. The distributed stochastic optimization over time-varying graphs with consensus projection procedure was studied in [32]. The consensus projection procedure made it possible to use more acceptable parameters of smoothness and strong convexity in the complexity estimates. The main goal of this paper is to investigate a similar consensus projection approach for problems with PL-condition. Note that approach [32, 33] is based on the well-known concept of the inexact (δ, L, μ)-oracle. But in the PL-case, we will use inexact gradients with additive noise to describe the consensus projection procedure.

1.2. Our contributions

Summing up, the contribution of this paper is as follows.

- We study the sum-type minimization problem when the objective function satisfies the PL-condition. To solve a such class of problems, we propose a gradient descent type method with consensus projection procedure (see Algorithm 2) and access to only inexact gradient. We consider two cases of gradient inexactness: the bounded deterministic inexactness and the sum of random noise with deterministic bias. We estimated the sufficient communication steps and iterations number to approach the required quality concerning the function and the distance between the agent's vectors and their average for both cases.

- We study the decentralized saddle-point problem (with the structure of a sum) when the objective function satisfies the two-sided PL-condition. For solving a such generalized class of problems, we proposed a generalization of the MGDA method (see Algorithm 3) with a consensus procedure. We provided an estimation for the sufficient number of iterations for inner and outer loops to approach the acceptable quality concerning the function. Also, we estimate the communication complexity to a distance between the agent's vectors and their average for both cases to be small enough. Additionally, we research the influence of

interference on the convergence of the proposed Algorithm 3. We suppose that the inexactness in the gradient can be separated into deterministic noise with the bounded norm and zero-mean noise with the finite second moment.

- We present some numerical experiments, which demonstrate the effectiveness of the proposed algorithms, for the Least Squares and Robust Least Squares problems.

2. Fundamentals and Assumptions for the problems under consideration

Throughout the paper, $\langle \cdot, \cdot \rangle$ denotes the inner product of vectors or matrices. Correspondingly, by $\| \cdot \|$, we denote the 2-norm for vectors or the Frobenius norm for matrices.

At the first, for the minimization problem (1) (or its equivalent (3)), we assume

Assumption 2.1 (Lipschitz smoothness). *For every $i = 1, \ldots, n$, the function f_i is L_i-smooth, for some $L_i > 0$.*

Under this assumption, we find that the function $F(\mathbf{X})$ (see (2)) is L_l-smooth on $\mathbb{R}^{n \times d}$, where $L_l = \max\limits_{1 \leq i \leq n} L_i$, and L_g-smooth on $\mathcal{C}$, where $L_g = \frac{1}{n} \sum\limits_{i=1}^{n} L_i$. The constant L_l is called a local constant and L_g is called a global constant.

Assumption 2.2 (PL-condition). *The function f satisfies the PL-condition, i.e., it holds the following inequality*

$$f(x) - f^* \leq \frac{1}{2\mu} \|\nabla f(x)\|^2, \quad \forall x \in \mathbb{R}^d, \tag{8}$$

for some $\mu > 0$, and f^* is the optimal value of the function f.

Also under this assumption, we find that f satisfies the quadratic growth condition (QG-condition) (see [15]):

$$\|x - x^*\|^2 \leq \frac{2}{\mu} \left(f(x) - f^* \right); \quad \forall x \in \mathbb{R}^d, \tag{9}$$

where x^* is the nearest point to the optimal solution of the minimization problem under consideration.

At the second, for the saddle-point problem (4), let us introduce the following assumption.

Assumption 2.3 (Lipschitz smoothness). *For every $i = 1, \ldots, n$, the function $\phi_i(\cdot, \cdot)$ is differentiable with respect to its both variables and smooth, i.e., the following inequalities hold*

$$\|\nabla_x \phi_i(x_1, y_1) - \nabla_x \phi_i(x_2, y_2)\| \leq L_{xx,i} \|x_1 - x_2\| + L_{xy,i} \|y_1 - y_2\|,$$
$$\|\nabla_y \phi_i(x_1, y_1) - \nabla_y \phi_i(x_2, y_2)\| \leq L_{yx,i} \|x_1 - x_2\| + L_{yy,i} \|y_1 - y_2\|,$$

for all $x_1, x_2 \in \mathbb{R}^{d_x}, y_1, y_2 \in \mathbb{R}^{d_y}$ and $L_{xx,i}, L_{yy,i}, L_{xy,i}, L_{yx,i} \in \mathbb{R}_+^*$.

Note, that if Assumption 2.3 is satisfied, then it will be also satisfied for the function

5

$\Phi(\cdot, \cdot)$ on $\mathbb{R}^{n \times d_x} \times \mathbb{R}^{n \times d_y}$ with constants $L_{xx,g}, L_{yx,g}, L_{xy,g}, L_{yy,g}$, and on $\mathcal{C}_x \times \mathcal{C}_y$, with constants $L_{xx,l}, L_{yx,l}, L_{xy,l}, L_{yy,l}$, where $L_{ab,l} = \max\limits_{1 \leq i \leq n} L_{ab,i}$, $L_{ab,g} = \frac{1}{n} \sum\limits_{i=1}^{n} L_{ab,i}$, and $ab \in \{xx, yy, xy, yx\}$.

2.1. Inexact gradient oracle

We will divide the inexactness of the gradient into random noise and a deterministic bias. To describe this noise, we will use the following definition.

Definition 2.4 $((\delta, \sigma^2)$-biased gradient oracle). Let $\delta > 0, \sigma > 0$ and ξ be a random variable with probability distribution $\mathcal{D}$. (δ, σ^2)-biased gradient oracle is a map $\mathbf{g}$: $\mathbb{R}^d \times \mathcal{D} \to \mathbb{R}^d$, such that

$$\|\mathbb{E}_\xi \mathbf{g}(x, \xi) - \nabla f(x)\| \leq \delta, \quad \mathbb{E}_\xi \|\mathbf{g}(x, \xi) - \mathbb{E}_\xi \mathbf{g}(x, \xi)\|^2 \leq \sigma^2.$$

In other words, we suppose that the inexactness of the gradient contains two parts: bias and noise. At that, bias has a bounded norm and noise has bounded the second moment.

2.2. Consensus procedure

We consider a sequence of non-directed communication graphs $\{\mathcal{G}^k = (V, E^k)\}_{k=0}^\infty$ and a sequence of corresponding mixing matrices $\{\mathbf{W}^k\}_{k=0}^\infty$ associated with it. We assume the following assumptions.

Assumption 2.5. *Mixing matrix sequence* $\{\mathbf{W}^k\}_{k=0}^\infty$ *satisfies the following properties:*
- *(Decentralized property) If* $(i, j) \notin E^k$*, then* $[\mathbf{W}^k]_{ij} = 0$.
- *(Double stochasticity)* $\mathbf{W}^k \mathbf{1}_n = \mathbf{1}_n$*, and* $\mathbf{1}_n^\top \mathbf{W}^k = \mathbf{1}_n^\top$.
- *(Contraction property) There exist* $\tau \in \mathbb{Z}_{++}$ *and* $\lambda \in (0, 1)$ *such that for every* $k \geq \tau - 1$*, it holds the following inequality*

$$\left\| \mathbf{W}_\tau^k \mathbf{X} - \overline{\mathbf{X}} \right\| \leq (1 - \lambda) \left\| \mathbf{X} - \overline{\mathbf{X}} \right\|,$$

where $\mathbf{W}_\tau^k = \mathbf{W}^k \ldots \mathbf{W}^{k-\tau+1}$.

The last property in Assumption (2.5) is a generalization of several well-known cases: time-static connected graph, sequence of the connected graph and τ-connected graph sequence [26]. A stochastic variant of this contraction property is also studied in [16].

During every communication round, the agents exchange information according to the rule

$$x_i^{k+1} = w_{ii}^k + \sum_{(i,j) \in E^k} w_{ij}^k x_j^k.$$

In matrix form, this update rule writes as $\mathbf{W}^{k+1} = \mathbf{W}^k \mathbf{X}^k$. The contraction property in Assumption (2.5) is needed to ensure geometric convergence of Algorithm 1 to the

average of nodes' initial vectors, i.e., to $\bar{x}_0$. In particular, the contraction property holds for τ-connected graphs with Metropolis weights choice for $\mathbf{W}^k$, i.e.,

$$
\left[\mathbf{W}^k\right]_{ij} = \begin{cases} 1/\left(1 + \max\{d_i^k, d_j^k\}\right) & \text{if } (i,j) \in E^k, \\ 0 & \text{if } (i,j) \notin E^k, \\ 1 - \sum_{(i,m)\in E^k} \left[\mathbf{W}^k\right]_{im} & \text{if } i = j, \end{cases} \tag{10}
$$

where d_i^k denotes the degree of node i in graph $\mathcal{G}^k$.

Algorithm 1 Consensus.

Require: Initial point $\mathbf{Z}^0 \in \mathcal{C}$, number of communication rounds T.
 1: Take current time moment t_0 from global variable.
 2: **for** $k = 0, \ldots, T - 1$ **do**
 3:    $\mathbf{Z}^{k+1} := \mathbf{W}^{t_0+k} \mathbf{Z}^k$.
 4: **end for**
 5: Update global variable with current time moment: $t_0 = t_0 + T$.
 6: **return** $\mathbf{Z}^T$.

Note, that the contraction property guarantee that after $T = N\tau$ steps of Algorithm 1 one obtains the point $\mathbf{Z}^0$ such that

$$
\|\mathbf{Z}^T - \overline{\mathbf{Z}}^0\| = \|W^{N\tau+t_0} \ldots W^{t_0}\mathbf{Z}^0 - \overline{\mathbf{Z}}^0\| \le (1 - \lambda)^N \|\mathbf{Z}^0 - \overline{\mathbf{Z}}^0\|.
$$

In other words, the consensus procedure converges with any accuracy because of contraction property in Assumption 2.5.

3. Algorithm for PL-minimization problem

In this section, we focus on the minimization problem 3 (which is equivalent to problem (1)). For this, we propose an algorithm (listed as Algorithm 2) using the inexact gradient. The proposed algorithm is a gradient-type method, that uses a $(\delta, 0)$-biased gradient oracle $\widetilde{\nabla} F(\mathbf{X})$ without noise. This condition can be rewritten as $\left\|\widetilde{\nabla} F(\mathbf{X}) - \nabla F(\mathbf{X})\right\| \le \delta$, where $\nabla F(\mathbf{X}) = (\nabla f_1(x_1), \ldots, \nabla f_n(x_n))^\top \in \mathbb{R}^{d\times n}$ denotes the gradient of F at $\mathbf{X}$. Note, that the considered inexact gradient in Algorithm 2 is a usual additively inexact gradient. Further, we note $\overline{\mathbf{X}}^k$ and $\overline{\widetilde{\nabla} F}(\mathbf{X}^k)$ as averaged $\mathbf{X}^k$ and $\widetilde{\nabla} F(\mathbf{X}^k)$ over consensus.

Algorithm 2 Decentralized gradient descent with consensus subroutine.

Require: Starting point $\mathbf{X}^0 \in \mathcal{C}$, step size $\gamma > 0$, number of steps N, the sequence of number for communication rounds $\{T^k\}_{k=0}^{N-1}$ in consensus.
 1: **for** $k = 0, \ldots, N - 1$ **do**
 2:    $\mathbf{Z}^{k+1} = \mathbf{X}^k - \gamma\widetilde{\nabla} F(\mathbf{X}^k)$,
 3:    $\mathbf{X}^{k+1} = \text{Consensus}(\mathbf{Z}^{k+1}, T^k)$.
 4: **end for**
 5: **return** $\mathbf{X}^N$.

Indeed, after the k-th iteration of Algorithm 2 we have a point $\mathbf{X}^{k+1} \approx \overline{\mathbf{X}}^{k+1} = \overline{\mathbf{X}}^k - \gamma \widetilde{\overline{\nabla}} F(\mathbf{X}^k)$. If we assume that at each iteration, $\mathbf{X}^k$ is close enough to its projection on $\mathcal{C}$, i.e., $\left\| \mathbf{X}^k - \overline{\mathbf{X}}^k \right\|^2 \leq \delta'$ for some $\delta' > 0$, then we can estimate the new inexactness. So, in the fact, it is a variant of gradient method for problem (3) with an additively inexact gradient. At the same time, $\overline{x}^{k+1} = \overline{x}^k - \gamma \widetilde{\overline{\nabla}} f(x^k)$ is the usual gradient method for minimizing the function f (i.e., for the problem (1)) with an additively inexact gradient $\widetilde{\overline{\nabla}} f(x^k) = \frac{1}{n} \sum\limits_{i=1}^{n} \widetilde{\nabla} f_i(x_i^k)$.

Let us introduce the following constant characterizes the distance from consensus space to points generated by Algorithm.

$$\sqrt{D} = \gamma \left\| \nabla F(\overline{\mathbf{X}}^*) \right\| + \sqrt{\delta'} + \left(\gamma + \frac{1}{\mu} \right) \Delta + \gamma L_g \sqrt{\frac{2}{\mu} \left(1 - \frac{\mu}{L_g} \right) \left(F(\overline{\mathbf{X}}^0) - F^* \right)}, \quad (11)$$

where $\gamma = \frac{1}{L_g}$. Using this value we can prove the following theorem.

Theorem 3.1. *Choose some $\varepsilon > 0$, define $D > 0$ as in (11), and set $\Delta = \delta + L_l \sqrt{\delta'}, \gamma = \frac{1}{L_g}$. Under Assumptions (2.1) and (2.2), Algorithm 2 requires*

$$N = \left\lceil \frac{L_g}{\mu} \log \left(\frac{f(\overline{x}^0) - f^*}{\varepsilon} \right) \right\rceil$$

gradient computation at each node, $T = \tau \left\lceil \frac{1}{2\lambda} \log \frac{D}{\delta'} \right\rceil$ communication steps at each iteration and $N_{tot} = N \cdot T$ communication steps to yield $\mathbf{X}^N$ such that

$$f(\overline{x}^N) - f^* \leq \varepsilon + \frac{\Delta^2}{2\mu n}, \quad and \quad \left\| \mathbf{X}^N - \overline{\mathbf{X}}^N \right\| \leq \sqrt{\delta'}. \quad (12)$$

Proof. Firstly, let us estimate the inexactness for inexact gradient $\widetilde{\overline{\nabla}} F(\mathbf{X}^k)$ in the following way:

$$\left\| \widetilde{\overline{\nabla}} F(\mathbf{X}^k) - \overline{\nabla F}(\overline{\mathbf{X}}^k) \right\| \leq \delta + L_l \sqrt{\delta'}.$$

Using this result we can obtain the similar estimate for $\widetilde{\overline{\nabla}} f(x^k) = \frac{1}{n} \sum\limits_{i=1}^{n} \widetilde{\nabla} f_i(x_i^k)$:

$$\left\| \widetilde{\overline{\nabla}} f(x^k) - \nabla f(\overline{x}^k) \right\|^2 = \frac{1}{n} \left\| \widetilde{\overline{\nabla}} F(\mathbf{X}^k) - \overline{\nabla F}(\overline{\mathbf{X}}^k) \right\|^2 = \frac{\left(\delta + L_l \sqrt{\delta'} \right)^2}{n}. \quad (13)$$

Using Assumption 2.2, inequality (13) and taking step size $\gamma = \frac{1}{L_g}$, we can estimate the convergence rate of Algorithm 2 with respect to the function f, as follows

$$f(\overline{x}^k) - f^* \leq \left(1 - \frac{\mu}{L_g} \right)^k \left(f(\overline{x}^0) - f^* \right) + \frac{\Delta^2}{2\mu n}, \quad (14)$$

where $\Delta = \delta + L_l\sqrt{\delta'}$. Although at each iteration of Algorithm 2 we have access only to $\mathbf{X}^k$.

Now, let us find the sufficient communication step such that the following expression is true

$$\left\| \mathbf{X}^k - \overline{\mathbf{X}}^k \right\| \leq \sqrt{\delta'} \Longrightarrow \left\| \mathbf{X}^{k+1} - \overline{\mathbf{X}}^{k+1} \right\| \leq \sqrt{\delta'}. \tag{15}$$

By the contraction property (see Assumption (2.5)) and using that $\overline{\mathbf{Z}}^{k+1} = \overline{\mathbf{X}}^{k+1}$, we have

$$\left\| \mathbf{X}^{k+1} - \overline{\mathbf{X}}^{k+1} \right\| \leq (1 - \lambda)^{\left\lfloor \frac{T_k}{\tau} \right\rfloor} \left\| \overline{\mathbf{X}}^{k+1} - \mathbf{Z}^{k+1} \right\|. \tag{16}$$

Let us estimate the right-hand side of inequality (16).

$$
\begin{aligned}
\left\| \overline{\mathbf{X}}^{k+1} - \mathbf{Z}^{k+1} \right\| &\leq \left\| \overline{\mathbf{X}}^k - \mathbf{Z}^{k+1} \right\| \leq \left\| \overline{\mathbf{X}}^k - \mathbf{X}^k \right\| + \gamma \left\| \widetilde{\nabla} F(\mathbf{X}^k) \right\| \\
&\leq \sqrt{\delta'} + \gamma \left\| \widetilde{\nabla} F(\mathbf{X}^k) - \nabla F(\overline{\mathbf{X}}^k) \right\| + \gamma \left\| \nabla F(\overline{\mathbf{X}}^k) - \nabla F(\overline{\mathbf{X}}^*) \right\| \\
&\quad + \gamma \left\| \nabla F(\overline{\mathbf{X}}^*) \right\| \\
&\leq \sqrt{\delta'} + \gamma\Delta + \gamma L_g \left\| \overline{\mathbf{X}}^k - \overline{\mathbf{X}}^* \right\| + \gamma \left\| \nabla F(\overline{\mathbf{X}}^*) \right\|.
\end{aligned} \tag{17}
$$

From quadratic growth condition (9), we have

$$\left\| \overline{\mathbf{X}}^k - \overline{\mathbf{X}}^* \right\|^2 = n \left\| \overline{x}^k - \overline{x}^* \right\|^2 \leq \frac{2n}{\mu} \left(f(\overline{x}^k) - f^* \right).$$

Further, using the convergence rate (14) we can estimate the value of $\left\| \overline{\mathbf{X}}^k - \overline{\mathbf{X}}^* \right\|^2$ in (17), as the following

$$\left\| \overline{\mathbf{X}}^k - \overline{\mathbf{X}}^* \right\|^2 \leq \frac{2}{\mu} \left(1 - \frac{\mu}{L_g} \right)^{k+1} \left(F(\overline{\mathbf{X}}^0) - F^* \right) + \frac{\Delta^2}{\mu^2}.$$

Therefore we have $\left\| \overline{\mathbf{X}}^{k+1} - \mathbf{Z}^{k+1} \right\| \leq \sqrt{D}$, where D is the constant defined according to 11

Thus, for $T_k = \tau \left\lceil \frac{1}{2\lambda} \log \frac{D}{\delta'} \right\rceil$, (15) will be satisfied. So, uniting (14) and the result about communication steps per iteration gets the theorem statement. $\qquad\square$

Further, we consider the case when Algorithm 2 uses (δ, σ^2)-biased gradient oracle $\widetilde{\nabla} F(\mathbf{X})$ for arbitrary values δ and σ.

As in the previous, after the k-th iteration of Algorithm 2 we have a point $\mathbf{X}^{k+1} \approx \overline{\mathbf{X}}^{k+1} = \overline{\mathbf{X}}^k - \gamma \overline{\widetilde{\nabla} F}(\mathbf{X}^k)$. In this case, if we consider $\overline{\widetilde{\nabla} F}(\mathbf{X}^k)$ as an approximation of the exact gradient $\overline{\nabla F}(\overline{\mathbf{X}}^k)$, then will have three inexactness: inexactness caused by $\overline{\mathbf{X}}^k \neq \mathbf{X}^k$, bias at point $\mathbf{X}^k$ and zero-mean noise at $\mathbf{X}^k$. Note, that $\mathbf{X}^k$ is also a random variable.

Let us estimate the bias in $\overline{\widetilde{\nabla} F}(\mathbf{X}^k)$ at the given point $\overline{\mathbf{X}}^k$, as follows

$$\sqrt{n}\left\|\mathbb{E}_{x^k,\xi}\overline{\widetilde{\nabla} f}(x^k) - \nabla f(\overline{x}^k)\right\| = \left\|\mathbb{E}_{\mathbf{X}^k,\xi}\overline{\widetilde{\nabla} F}(\mathbf{X}^k) - \overline{\nabla F}(\overline{\mathbf{X}}^k)\right\|$$

$$\leq \left\|\mathbb{E}_{\mathbf{X}^k,\xi}\overline{\widetilde{\nabla} F}(\mathbf{X}^k) - \mathbb{E}_{\mathbf{X}^k}\overline{\nabla F}(\mathbf{X}^k)\right\| + \left\|\mathbb{E}_{\mathbf{X}^k}\overline{\nabla F}(\mathbf{X}^k) - \overline{\nabla F}(\overline{\mathbf{X}}^k)\right\|$$

$$\leq \mathbb{E}_{\mathbf{X}^k}\left\|\mathbb{E}_\xi\overline{\widetilde{\nabla} F}(\mathbf{X}^k) - \overline{\nabla F}(\mathbf{X}^k)\right\| + \mathbb{E}_{\mathbf{X}^k}\left\|\overline{\nabla F}(\mathbf{X}^k) - \overline{\nabla F}(\overline{\mathbf{X}}^k)\right\|$$

$$\leq \delta + L_l\mathbb{E}_{\mathbf{X}^k}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|,$$

where $\mathbb{E}_{\mathbf{X}^k}$ and $\mathbb{E}_\xi$ mean conditional mathematical expectations under variable $\mathbf{X}^k$ and random variable ξ for the given $\overline{\mathbf{X}}^k$. As a result, one requires $\mathbb{E}_{\mathbf{X}^k}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta'$ for small bias of gradient. On the other hand, we can construct Algorithm 2 in such a way that $\mathbb{E}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta'$ at all iterations. When we assume, that Consensus procedure 1 guarantees $\mathbb{E}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta'$, we have the following estimation

$$\mathbb{E}\left\|\mathbb{E}_{x,\xi}\overline{\widetilde{\nabla} f}(x^k) - \nabla f(\overline{x}^k)\right\|^2 \leq \frac{2\delta^2 + 2L_l^2\delta'}{n}.$$

At the same time, we can estimate the random noise in a similar way (see Appendix A) and it gets the following lemma.

Lemma 3.2. *Let us assume that the functions f_i meet Assumption 2.1 and for all $j \leq k$ we have that $\mathbb{E}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\|^2 \leq \delta'$, where $\{\mathbf{X}^j\}_{j \leq k}$. Then for the bias and noise in the inexact gradient $\overline{\widetilde{\nabla} f}(x^k)$, we have the following estimations*

$$\mathbb{E}\left\|\mathbb{E}_{x,\xi}\overline{\widetilde{\nabla} f}(x^k) - \nabla f(\overline{x}^k)\right\|^2 \leq \frac{2\delta^2 + 2L_l^2\delta'}{n},$$

and

$$\mathbb{E}\left\|\mathbb{E}_{x,\xi}\overline{\widetilde{\nabla} f}(x^k) - \overline{\widetilde{\nabla} f}(x^k)\right\|^2 \leq \frac{16L_l^2\delta' + 18\sigma^2 + 16\delta^2}{n}.$$

Note that the mathematical expectation $\mathbb{E}$, above is not conditional. So, $\overline{\widetilde{\nabla} f}(x^k)$ is not a biased in the sense of Definition 2.4. Nevertheless, we can use the following gradient method

$$x_k = x_{k-1} - \gamma g(x_k, \xi), \quad k = 1, 2, \ldots \tag{18}$$

where $g(x_k, \xi) = \nabla f(x_k) + n(x_k, \xi) + b(x^k)$ such that $\mathbb{E}\|n(x_k, \xi)\|^2 \leq \sigma^2$ and $\mathbb{E}\|b(x_k)\|^2 \leq \delta^2$. The convergence of such a method is given by the following lemma (see Lemma 2 in [1]).

Lemma 3.3. *Let f be a function that satisfies the PL-condition for a constant $\mu > 0$ and be an L-smooth function. The gradient oracle $g(x_k, \xi) = \nabla f(x_k) + n(x_k, \xi) + b(x^k)$ is such that $\mathbb{E}\|n(x_k, \xi)\|^2 \leq \sigma^2$ and $\mathbb{E}\|b(x_k)\|^2 \leq \delta^2$. When $\gamma \leq \frac{1}{L}$, we can guarantee*

that method (18) *converges to* f^* *in the following way*

$$\mathbb{E}\left[f(x_k) - f^*\right] \leq (1 - \gamma\mu)^k \left(f(x_0) - f^*\right) + \frac{\delta^2}{2\mu} + \frac{L\gamma\sigma^2}{2\mu}. \tag{19}$$

In the similar way, like for proof of Theorem 3.1, we can estimate required consensus steps T for condition $\mathbb{E}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta'$ (see Appendix C). So, we can state the following theorem.

Theorem 3.4. *Let f be a function meets Assumption 2.2, the functions f_i meet Assumption 2.1 and the map $\widetilde{\nabla}F$ be a (δ, σ^2)-biased gradient oracle. Define value D as*

$$D = 6\gamma^2 \left\|\nabla F(\overline{\mathbf{X}}^*)\right\|^2 + 2\delta' + 6\left(\gamma^2 + \frac{1}{\mu^2}\right)\Delta^2 + \frac{12\gamma^2 L_g^2}{\mu}\left(1 - \frac{\mu}{L_g}\right)\left(F(\overline{\mathbf{X}}^0) - F^*\right),$$

where $\Delta^2 = 18\left(L_l^2\delta' + \sigma^2 + \delta^2\right)$ and step-size $\gamma = \frac{1}{L_g}$. Then Algorithm 2 requires $N = \left\lceil \frac{L_g}{\mu}\log\left(\frac{f(\overline{x}^0) - f^*}{\varepsilon}\right)\right\rceil$ gradient computation at each node, $T = \tau\left\lceil \frac{1}{2\lambda}\log\frac{D}{\delta'}\right\rceil$ communication steps in each iteration and $N_{tot} = N \cdot T$ communication steps to yield $\mathbf{X}^N$ such that

$$\mathbb{E}f(\overline{x}^N) - f^* \leq \varepsilon + \frac{\Delta^2}{2\mu n}, \quad and \quad \mathbb{E}\left\|\mathbf{X}^N - \overline{\mathbf{X}}^N\right\|^2 \leq \delta'. \tag{20}$$

Remark 1. By using the step-size $\gamma < \frac{1}{L_g}$ small enough, we can improve the accuracy level on function in (20). According to Lemma 3.3, for step-size γ, Algorithm 2 can converge with the rate as in the inequality (20), where

$$\Delta^2 = 2\delta^2 + L_l^2\delta' + L_g\gamma\left(16L_l^2\delta' + 18\sigma^2 + 16\delta^2\right).$$

But in this case the number of required iterations N increases in $\frac{L_g}{\gamma}$ times. Note, that parameter D does not change.

Remark 2. Note, that the convergence rate is almost optimal for functions with PL-condition (see [43]).

4. Algorithm for distributed saddle point problems: PL–PL case

In this section, we will consider a generalization of Algorithm 2 (see Algorithm 3) for the saddle-point problem (4) (or its equivalent (6)).

Additionally, we will research the influence of the inexact access to the oracle (especially using the inexact information of the gradient) on the convergence of the proposed Algorithm 3. We suppose, that the inexactness in the gradient can be separated into deterministic noise with the bounded norm and zero-mean noise with the finite second moment. Note, that the influence of such inexactness for Gradient Descent Ascent is well researched both in convex-concave and in non-convex-non-concave saddle point problems under various assumptions (see e.g. [41], [5]). Also, the results of [25] can be generalized for stochastic oracle (see Remark 3.8 in [25]).

11

At each iteration, the proposed method optimizes by an inner variable $\mathbf{Y}$ for the fixed outer variable $\mathbf{X}$ by Algorithm 2. After that, it makes one step of Algorithm 3 by outer variable. So, we obtain the well-known Multi-step Gradient Descent Ascent method with a consensus subroutine after each gradient method step.

Algorithm 3 Multi-step Gradient Descent Ascent with consensus (MGDA).

Require: Number of the outer and inner steps N_x, N_y, starting point $(\mathbf{X}^0, \mathbf{Y}^0)$, step sizes γ_x and γ_y, the sequences of steps $\{T_x^k\}_k$ and $\{T_y^{k,j}\}_{k,j}$ in consensus.
1: **for** $k = 0, \ldots, N_x - 1$ **do**
2:     $\hat{\mathbf{Y}}^0 := \mathbf{Y}^k$.
3:     **for** $j = 0, \ldots, N_y - 1$ **do**
4:        $\mathbf{Z}_y^{j+1} := \hat{\mathbf{Y}}^j + \gamma_y \widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j)$.
5:        $\hat{\mathbf{Y}}^{j+1} := \text{Consensus}(\mathbf{Z}_y^{j+1}, T_y^{k,j})$.
6:     **end for**
7:     $\mathbf{Y}^{k+1} := \hat{\mathbf{Y}}^{N_y}$.
8:     $\mathbf{Z}_x^{k+1} := \mathbf{X}^k - \gamma_x \widetilde{\nabla}_{\mathbf{X}} \Phi(\mathbf{X}^k, \mathbf{Y}^{k+1})$.
9:     $\mathbf{X}^{k+1} := \text{Consensus}(\mathbf{Z}_x^{k+1}, T_x^k)$.
10: **end for**
11: **return** $\mathbf{X}^{N_x}, \mathbf{Y}^{N_x}$.

We suppose that functions $\phi(\cdot, y)$ and $-\phi(x, \cdot)$ satisfy the PL-condition (Assumption 2.2). So, for each x and y the following inequalities hold

$$\phi(x, y) - \min_{x \in \mathbb{R}^{d_x}} \phi(x, y) \leq \frac{2}{\mu_x} \|\nabla_x \phi(x, y)\|^2, \tag{21}$$

$$\max_{y \in \mathbb{R}^{d_y}} \phi(x, y) - \phi(x, y) \leq \frac{2}{\mu_y} \|\nabla_y \phi(x, y)\|^2. \tag{22}$$

Now, let us introduce functions in $\mathbf{Y}$ for fixed variable $\mathbf{X}$: $g_{\overline{x}}(y) = \phi(\overline{x}, y)$ and $G_{\overline{\mathbf{X}}}(\mathbf{Y}) = \Phi(\overline{\mathbf{X}}, \mathbf{Y})$. For the maximization of $g_{\overline{x}}(y)$ we define $g_x^* := \max_{y \in \mathbb{R}^{d_y}} g_{\overline{x}}(y)$.

Additionally, we assume that there are points for functions F and $G_{\overline{X}}$ in consensus subspace minimizing this functions in the full space. In other words, the following conditions hold:

$$\forall \overline{X} \; \exists \overline{Y}^* \in \mathcal{C}_y : \max_{Y \in \mathbb{R}^{n \times d_y}} G_{\overline{X}}(Y) = G_{\overline{X}}(\overline{Y}^*), \tag{23}$$

and

$$\exists \overline{X}^* \in \mathcal{C}_x : \min_{X \in \mathbb{R}^{n \times d_x}} F(X) = F(\overline{X}^*). \tag{24}$$

This conditions allows to obtain that Assumption 2.2 holds for full space for any subproblem.

Note, in the inner iterations we have gradient process with respect to $\mathbf{Y}$. Uniting this assumption and results of the previous part we obtain the following result.

Lemma 4.1. *Let us define* $\Delta_y^2 = \delta + L_{yy,l} \sqrt{\delta_y'} + L_{yx,l} \sqrt{\delta_x'}$ *and* D_y *by the following*

way

$$\sqrt{D_{\mathbf{X},\mathbf{Y}}} = \gamma_y \left\| \nabla G_{\overline{x}}(\overline{\mathbf{Y}}^*) \right\| + \sqrt{\delta_y'} + \left(\gamma_y + \frac{1}{\mu_y} \right) \Delta_y$$
$$+ \gamma_y L_{yy,g} \sqrt{\frac{2n}{\mu_y} \left(1 - \frac{\mu_y}{L_{yy,g}} \right) \left(G_{\overline{x}}(\overline{\mathbf{Y}}) - G_{\overline{x}}^* \right)}. \tag{25}$$

Besides let us method makes at least $N_y = \left\lceil \frac{L_{yy,g}}{\mu} \log \frac{g_x(\overline{y}^0) - g_x^*}{\varepsilon} \right\rceil$ *iterations of inner loop for* $T_y = \tau \left\lceil \frac{1}{2\lambda} \log \left(\frac{D_{y,x}}{\delta_y'} \right) \right\rceil$ *communication steps. If Assumption 2.2 and statement (23) hold, the method obtains a point* $\mathbf{Y}^k$ *for any outer iteration* k *such that*

$$\max_{y \in \mathbb{R}^{d_y}} \phi(\overline{x}^k, \overline{y}^k) - \phi(\overline{x}^k, \overline{y}^k) \leq \varepsilon_y + \frac{\Delta_y^2}{2\mu_y n} = \hat{\varepsilon}_y, \quad and \quad \left\| \mathbf{Y}^k - \overline{\mathbf{Y}}^N \right\| \leq \sqrt{\delta_y'},$$

Proof. Note, that in the inner iterations, we have the gradient process in the form

$$\hat{\mathbf{Y}}^j \approx \overline{\hat{\mathbf{Y}}}^j = \overline{\hat{\mathbf{Y}}}^{k-1} + \gamma_y \widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j).$$

Let $\widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}, \mathbf{Y})$ be a $(\delta_y, 0)$-biased gradient oracle for any fixed $\mathbf{X}$. So, we can decompose the bias component in $\widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j)$ in the following way

$$\left\| \widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j) - \nabla_y \Phi(\overline{\mathbf{X}}^k, \overline{\hat{\mathbf{Y}}}^j) \right\| \leq \left\| \widetilde{\nabla}_{\mathbf{Y}} \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j) - \nabla_y \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j) \right\|$$
$$+ \left\| \nabla_y \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j) - \nabla_y \Phi(\mathbf{X}^k, \overline{\hat{\mathbf{Y}}}^j) \right\| \tag{26}$$
$$+ \left\| \nabla_y \Phi(\mathbf{X}^k, \overline{\hat{\mathbf{Y}}}^j) - \nabla_y \Phi(\overline{\mathbf{X}}^k, \overline{\hat{\mathbf{Y}}}^j) \right\|.$$

The first term on the right-hand side of inequality (26) is not more than δ. Assuming that $\left\| \overline{\hat{\mathbf{Y}}}^j - \hat{\mathbf{Y}}^j \right\|^2 \leq \delta_y'$ and $\left\| \overline{\mathbf{X}}^k - \mathbf{X}^k \right\|^2 \leq \delta_x'$ for any k, j, we can estimate the last two terms in (26) as $L_{yy} \sqrt{\delta_y'} + L_{yx} \sqrt{\delta_x'}$. So, we have the following estimate for bias in the inexact gradient $\overline{\widetilde{\nabla}_{\mathbf{Y}} \phi}(\mathbf{X}^k, \hat{\mathbf{Y}}^j) = \frac{1}{n} \sum_{i=1}^{n} \phi(x_{k,i}, \hat{y}_i^j)$:

$$\left\| \overline{\widetilde{\nabla}_{\mathbf{Y}} \phi}(\mathbf{X}^k, \hat{\mathbf{Y}}^j) - \nabla_y \phi(\overline{x}_k, \overline{\hat{y}}^j) \right\| \leq \frac{1}{n} \left\| \widetilde{\nabla}_Y \Phi(\mathbf{X}^k, \hat{\mathbf{Y}}^j) - \nabla_y \Phi(\overline{\mathbf{X}}^k, \overline{\hat{\mathbf{Y}}}^j) \right\|$$
$$\leq \frac{(\delta + L_{yy,l} \sqrt{\delta_y'} + L_{yx,l} \sqrt{\delta_x'})^2}{n}.$$

Finally, note, that g_x is $L_{yy,g}$ smooth function. So, the convergence by $\mathbf{Y}$ is described by Theorem 3.4. So, we have the result of this lemma. $\square$

Note, in not decentralized case we have that the inexact gradient is close enough to exact one: $\overline{\nabla_{\mathbf{X}} \Phi}(\overline{\mathbf{X}}^k, \mathbf{Y}^k) \approx \overline{\nabla_{\mathbf{X}} F}(\overline{\mathbf{X}}^k) = \overline{\nabla_{\mathbf{X}} \Phi}(\overline{\mathbf{X}}^k, \overline{\mathbf{Y}}^*)$ (see Lemma A.6 from [25]).

This case corresponds to every full-connected graph. Let us generalize this result for the decentralized problem statement.

Theorem 4.2. *Let functions $\phi(\cdot, y)$ and $-\phi(x, \cdot)$ meet Assumption 2.2 with constants $\mu_x > 0$ and $\mu_y > 0$, and functions ϕ_i meet the Assumption 2.3. Besides statements (23) and (24) hold. The gradient oracle $(\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X}, \mathbf{Y}), \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X}, \mathbf{Y}))$ be $(\delta, 0)$-biased gradient oracle. Also, Assumption 2.5 holds. Let us introduce inexactness values $\Delta_y := \delta + L_{yy,l}\sqrt{\delta_y'} + L_{yx,l}\sqrt{\delta_x'}$ and $\Delta_x = \delta + L_{xx,l}\sqrt{\delta_x'} + L_{xy,l}\left(\sqrt{\frac{\varepsilon_y}{2\mu_y}} + \frac{\Delta_y}{2\mu_y\sqrt{n}} + \sqrt{\delta_y'}\right)$. Also define values $D_{\mathbf{X}}$ according to (29) and $D_{\mathbf{Y}} = \max_k D_{\mathbf{X}^k, \mathbf{Y}}$, where $D_{\mathbf{X}^k, \mathbf{Y}}$ defined in (25).*

Let Algorithm 3 makes $T_x = \tau\left\lceil \frac{1}{2\lambda}\log\left(\frac{D_{\mathbf{X}}}{\delta_x'}\right)\right\rceil$ communication steps in each outer iteration and $T_y = \tau\left\lceil \frac{1}{2\lambda}\log\left(\frac{D_{\mathbf{Y}}}{\delta_y'}\right)\right\rceil$ communication steps at each inner iteration, with step-sizes $\gamma_x = \frac{1}{L_x}$ and $\gamma_y = \frac{1}{L_{yy,g}}$. Then Algorithm 3 requires

$$N_x = \left\lceil \frac{L_x}{\mu_x}\log\left(\frac{F(\overline{\mathbf{X}}^0) - F^*}{\varepsilon_x}\right)\right\rceil$$

outer iterations, where $L_x = L_{xx,g} + \frac{L_{xy,g}}{\mu_y}$, and

$$N_y = \left\lceil \frac{L_{yy,g}}{\mu_y}\log\left(\frac{G_{\overline{x}}^* - G_{\overline{x}}(\overline{\mathbf{X}}^0)}{\varepsilon_y}\right)\right\rceil$$

inner iterations for each outer iteration to yield the pair $\left(\mathbf{X}^{N_x}, \mathbf{Y}^{N_y}\right)$, such that

$$f(\overline{x}^{N_x}) - f^* \leq \varepsilon_x + \frac{\Delta_x^2}{2\mu_x n}, \quad \left\|\mathbf{X}^{N_x} - \overline{\mathbf{X}}^{N_x}\right\| \leq \sqrt{\delta_x'}, \tag{27}$$

and

$$\max_{y \in \mathbb{R}^{d_y}} \phi(\overline{x}^{N_x}, y) - \phi(\overline{x}^{N_x}, \overline{y}^{N_y}) \leq \varepsilon_y + \frac{\Delta_y^2}{2\mu_y n}, \quad \left\|\mathbf{Y}^{N_y} - \overline{\mathbf{Y}}^{N_y}\right\| \leq \sqrt{\delta_y'}. \tag{28}$$

Proof. We proved, that after N_y iterations of inner loop, we have such point $Y = \hat{\mathbf{Y}}^{N_y}$ that $g_x^* - g_x(\overline{y}) \leq \varepsilon_y$ and $\left\|\mathbf{Y} - \hat{\mathbf{Y}}\right\| \leq \sqrt{\delta_y'}$. Using QG-condition for the function g_x, we can estimate the distance from obtained point to the optimal one as $\left\|\mathbf{Y} - \overline{\mathbf{Y}}^*(\mathbf{X})\right\| \leq \sqrt{\frac{\hat{\varepsilon}_y}{2\mu_y}} + \sqrt{\delta_y'}$, where $\overline{y}^*(x) = \arg\max_{y \in \mathbb{R}^{d_y}} g_{\overline{x}}$. So, we can estimate the inexactness of the gradient with respect to $\mathbf{X}$ at the point $\mathbf{Y} = \hat{\mathbf{Y}}^{N_y}$ according to the Assumption

2.3 in the following way

$$\left\|\widetilde{\overline{\nabla}_{\mathbf{X}}\Phi}(\mathbf{X},\mathbf{Y}) - \overline{\nabla F}(\overline{\mathbf{X}})\right\| \leq \delta + L_{xx,l}\left\|\mathbf{X} - \overline{\mathbf{X}}\right\| + L_{xy,l}\left\|\mathbf{Y} - \overline{\mathbf{Y}}^*(\overline{\mathbf{X}})\right\|$$

$$\leq \delta + L_{xx,l}\sqrt{\delta_x'} + L_{xy,l}\left(\sqrt{\frac{\hat{\varepsilon}_y}{2\mu_y}} + \sqrt{\delta_y'}\right)$$

$$\leq \delta + L_{xx,l}\sqrt{\delta_x'} + L_{xy,l}\left(\sqrt{\frac{\varepsilon_y}{2\mu_y}} + \frac{\Delta_y}{2\mu_y\sqrt{n}} + \sqrt{\delta_y'}\right).$$

So, through the inner iterations, we obtain an inexact gradient with respect to x such that $\left\|\widetilde{\nabla}f(x) - \nabla f(\overline{x})\right\| \leq \frac{\Delta_x}{\sqrt{n}}$, where

$$\Delta_x = \delta + L_{xx,l}\sqrt{\delta_x'} + L_{xy,l}\left(\sqrt{\frac{\varepsilon_y}{2\mu_y}} + \frac{\Delta_y}{2\mu_y\sqrt{n}} + \sqrt{\delta_y'}\right).$$

At the same time, the function $\max_{y\in\mathbb{R}^{d_y}}\phi(x,y)$ meets PL-condition. Moreover, as known (see Lemma A.3 in [41]), when the objective function ϕ meets Assumptions 2.3 and condition (21), the function $\max_{y\in\mathbb{R}^{d_y}}\phi(x,y)$ meets PL-condition with constant μ_x until if $\min_x\max_{y\in\mathbb{R}^{d_y}}\phi(x,y) = \min_{\mathbf{X}}F(\mathbf{X})$.. Also note, that under Assumptions 2.3 and PL-condition (22) for inner variable, the function $F(\mathbf{X}) = \Phi(\mathbf{X},\mathbf{Y}^*(\mathbf{X}))$ is an L_x-Lipschitz continuous function with $L_x = L_{xx,g} + \frac{L_{xy,g}}{\mu_y}$ (see Lemma A.2 in [41]). So, from Theorem 3.1, we obtain that Algorithm 3 requires $N_x = \left\lceil\frac{L_x}{\mu_x}\log\left(\frac{F(\overline{\mathbf{X}}^0)-F^*}{\varepsilon_x}\right)\right\rceil$ outer iterations and $T_x = \tau\left\lceil\frac{1}{2\lambda}\log\left(\frac{D_{\mathbf{X}}}{\delta_x'}\right)\right\rceil$ where

$$\sqrt{D_{\mathbf{X}}} = \gamma_x\left\|\nabla F(\overline{\mathbf{X}}^*)\right\| + \sqrt{\delta_x'} + \left(\gamma_x + \frac{1}{\mu_x}\right)\Delta_x + \gamma_x L_x\sqrt{\frac{2}{\mu_x}\left(1 - \frac{\mu_x}{L_x}\right)\left(F(\overline{\mathbf{X}}^0) - F^*\right)}.$$

(29)

Uniting obtained above results for convergences by $\mathbf{X}$ and $\mathbf{Y}$, we obtain the result of the theorem. $\qquad\square$

Remark 3. In general, Algorithm 3 requires

$$N_x \cdot N_y = O\left(\left(\frac{L_{xx,g}L_{yy,g}}{\mu_y\mu_x} + \frac{L_{xy,g}L_{yy,g}}{\mu_y^2\mu_x}\right)\log^2\frac{1}{\varepsilon}\right),$$

gradient computations with respect to $\mathbf{Y}$ and

$$N_x = O\left(\left(\frac{L_{xx,g}}{\mu_x} + \frac{L_{xy,g}}{\mu_y\mu_x}\right)\log\frac{1}{\varepsilon}\right),$$

gradient computations with respect to $\mathbf{X}$ at each node.

Remark 4. Algorithm 3 requires $T_{tot} = N_xT_x + N_yN_xT_y$ communication steps to achieve the required quality. Note, that the communication steps in inner iterations and outer iterations can have different computational costs. Namely, the dimensions d_x for $\mathbf{X}$ and d_y for $\mathbf{Y}$ can significantly differ.

Remark 5. The main restriction in the obtained result is conditions 23 and 24 and PL-condition. Nevertheless, note that such situation is typical for overparametrized problems that includes many problems from modern deep learning (see [4], [15]).

Remark 6. The particular case of functions with PL-condition is strong convexity. In such case, the result can be significantly accelerated (see e.g. [33]).

Now, we consider the case, when $\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X}, \mathbf{Y})$ and $\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X}, \mathbf{Y})$ be (δ, σ^2)-biased gradient oracle. Let parameters numbers of communication steps T_x and T_y such that $\mathbb{E}\|\mathbf{X}^j - \overline{\mathbf{Y}}^j\|^2 \leq \delta'_x$ and $\mathbb{E}\|\mathbf{Y}^j - \overline{\mathbf{Y}}^j\|^2 \leq \delta'_y$ in all iterations. So, we can estimate noise and bias for the gradient with respect to $\mathbf{Y}$ for each inner loop. The proof of the following lemma is presented in Appendix D.

Lemma 4.3. *Let us assume that functions ϕ meet Assumption 2.3 and for all $j \leq k$ we have $\mathbb{E}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\|^2 \leq \delta'_x$, $\mathbb{E}\left\|\hat{\mathbf{Y}}^j - \overline{\hat{\mathbf{Y}}}^j\right\|^2 \leq \delta'_y$, where $\{\mathbf{X}^j\}_{j \leq k}$ and $\{\hat{\mathbf{Y}}^j\}_{j \leq k}$. Then for the bias and noise in the inexact gradient $g_{k,j} = \overline{\widetilde{\nabla}_Y \phi}(\mathbf{X}^k, \mathbf{Y}^k) = \frac{1}{n}\sum\limits_{i=1}^{m} \widetilde{\nabla}_y \phi_i(x_{k,i}, y_{k,i})$ we the following estimations:*

$$\mathbb{E}\left\|\mathbb{E}_{X,Y,\xi} g_{k,j} - \nabla_Y \phi(\overline{x}^k, \overline{y}^j)\right\|^2 \leq \frac{3\delta^2 + 3L_{yy}^2 \delta'_y + 3L_{yx}^2 \delta'_x}{n},$$

and

$$\mathbb{E}\left\|\mathbb{E}_{X,Y,\xi} g_{k,j} - g_{k,j}\right\|^2 \leq 16\left(\frac{L_{yy,l}^2 \delta'_y + L_{yx,l}^2 \delta'_x + \delta^2}{n}\right) + \frac{18\sigma^2}{n}.$$

It allows us to use the results of Theorem 3.4. Let parameters N_y and T_y be defined according to this theorem for the current value of outer variable $\mathbf{X}^k$ and chosen accuracies ε_y and δ'_y. Then after N_y inner iterations we obtain such point $\mathbf{Y} = \hat{\mathbf{Y}}^{N_y}$, s.t.

$$\max_{y \in \mathbb{R}^{d_y}} \phi(\overline{x}^k, y) - \mathbb{E}\phi(\overline{x}^k, \overline{y}^{N_y}) \leq \varepsilon_y + \frac{\Delta_y^2}{2\mu_y n}, \quad \text{and} \quad \mathbb{E}\left\|\hat{\mathbf{Y}}^{N_y} - \overline{\hat{\mathbf{Y}}}^{N_y}\right\|^2 \leq \delta'_y,$$

where $\Delta_y^2 = 19\left(L_{yy,l}^2 \delta'_y + L_{yx,l}^2 \delta'_x + \sigma^2 + \delta^2\right)$.

This allows us to estimate inaccuracies in the inexact gradient with respect to $\mathbf{X}$ on outer iterations. Note, that it contains new inexactness because of the inexact solution of the inner problem on each iteration. The following lemma contains results about the gradient bias and noise (see the proof in Appendix E).

Lemma 4.4. *Let us assume that the function ϕ meets Assumption 2.3 and for all $j \leq k$ we have $\mathbb{E}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\|^2 \leq \delta'_x$, $\mathbb{E}\left\|\mathbf{Y}^j - \overline{\mathbf{Y}}^j\right\|^2 \leq \delta'_y$, where $\{\mathbf{X}^j\}_j$ and $\{\mathbf{Y}^j\}_j$ are the sequence generated by Algorithm 3. Also, $\mathbf{Y}^k$ be a point such that $g_x(\overline{y}) - g_x^* \leq \varepsilon_y + \frac{\Delta_y^2}{2\mu_y n}$. Then for the bias and noise in the inexact gradient $h_k = \overline{\widetilde{\nabla}_{\mathbf{Y}}\phi}(\mathbf{X}^k, \mathbf{Y}^k) = \frac{1}{n}\sum\limits_{i=1}^{n} \widetilde{\nabla}_y \phi_i(x_{k,i}, y_{k,i})$, we the following estimations*

16

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}h_k - \nabla F(\overline{\mathbf{Y}})\right\|^2 \leq \frac{3\delta^2 + 3L_{xx}^2\delta_x' + 6L_{xy}^2\left(\frac{2\varepsilon_y}{\mu_y} + \frac{\Delta_y^2}{\mu_y^2 n} + \delta_y'\right)}{n},$$

and

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},Y,\xi}h_k - h_k\right\|^2 \leq 16\frac{L_{xy,l}^2\delta_y' + L_{xx,l}^2\delta_x' + \delta^2}{n} + \frac{18\sigma^2}{n}.$$

Uniting the results of Lemmas 4.3, 4.4 and Theorem 3.4, we obtain the following result for the convergence of Algorithm 3.

Theorem 4.5. *Let the functions $\phi(\cdot, y)$ and $-\phi(x, \cdot)$ meet Assumption 2.2 with constants $\mu_x > 0$ and $\mu_y > 0$ and the functions ϕ_i meet the Assumption 2.3. Besides statements (23) and (24) hold. The gradient oracle $\left(\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}), \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right)$ be a (δ, σ^2)-biased gradient oracle. Also, Assumption 2.5 holds. Let us introduce the following inexactness values*

$$\Delta_y^2 := 19\left(L_{yy,l}^2\delta_y' + L_{yx,l}^2\delta_x' + \sigma^2 + \delta^2\right),$$

$$\Delta_x^2 := 22L_{xy,l}^2\delta_y' + 19L_{xx,l}^2\delta_x' + 19\delta^2 + 18\sigma^2 + 6L_{xy}^2\left(\frac{2\varepsilon_y}{\mu_y} + \frac{\Delta_y^2}{\mu_y^2 n}\right).$$

and define the values $D_{\mathbf{X}}$ and $D_{\mathbf{Y}}$, such that

$$D_{\mathbf{X}} = 6\gamma_x^2\left\|\nabla F(\overline{\mathbf{X}}^*)\right\|^2 + 2\delta_x' + 6\left(\gamma_x^2 + \frac{1}{\mu_x^2}\right)\Delta^2 + \frac{12\gamma_x^2 L_x^2}{\mu_x}\left(1 - \frac{\mu_x}{L_x}\right)\left(F(\overline{\mathbf{X}}^0) - F^*\right),$$

and $D_{\mathbf{Y}} = \max_k D_{\mathbf{X}^k,\mathbf{Y}}$, where

$$D_{\mathbf{X}^k,\mathbf{Y}} = 6\gamma^2\mathbb{E}\left\|\nabla G_{\mathbf{X}^k}(\overline{\mathbf{Y}}^*)\right\|^2 + 2\delta_y' + 6\left(\gamma_y^2 + \frac{1}{\mu_y^2}\right)\Delta_y^2$$

$$+ \frac{12\gamma^2 L_{yy,g}^2}{\mu_y}\left(1 - \frac{\mu_y}{L_{yy,g}}\right)\mathbb{E}\left[G_{\mathbf{X}^k}(\overline{\mathbf{Y}}^0) - G_{\mathbf{X}}^*\right],$$

Let Algorithm 3, with step-sizes $\gamma_x = \frac{1}{L_x}, \gamma_y = \frac{1}{L_{yy,g}}$, makes $T_x = \tau\left\lceil\frac{1}{2\lambda}\log\left(\frac{D_{\mathbf{X}}}{\delta_x'}\right)\right\rceil$ communication steps in each outer iteration and $T_y = \tau\left\lceil\frac{1}{2\lambda}\log\left(\frac{D_{\mathbf{Y}}}{\delta_y'}\right)\right\rceil$ communication steps in each inner iteration. Then Algorithm 3 requires

$$N_x = \left\lceil\frac{L_x}{\mu_x}\log\left(\frac{F(\overline{\mathbf{X}}^0) - F^*}{\varepsilon_x}\right)\right\rceil$$

outer iterations, where $L_x = L_{xx,g} + \frac{L_{xy,g}}{\mu_y}$, and

$$N_y = \left\lceil\frac{L_{yy,g}}{\mu_y}\log\left(\frac{G_{\overline{x}}^* - G_{\overline{x}}(\overline{\mathbf{X}}^0)}{\varepsilon_y}\right)\right\rceil$$

17

inner iterations for each outer iteration to yield the pair $(\mathbf{X}^{N_x}, \mathbf{Y}^{N_y})$ *such that*

$$\mathbb{E}\left[f\left(\overline{x}^{N_x}\right) - f^*\right] \leq \varepsilon_x + \frac{\Delta_x^2}{2\mu_x n}, \quad \mathbb{E}\left\|\mathbf{X}^{N_x} - \overline{\mathbf{X}}^{N_x}\right\| \leq \sqrt{\delta_x'}, \tag{30}$$

and

$$\mathbb{E}\left[\max_{y \in \mathbb{R}^{d_y}} \phi\left(\overline{x}^{N_x}, y\right) - \phi\left(\overline{x}^{N_x}, \overline{y}^{N_y}\right)\right] \leq \varepsilon_y + \frac{\Delta_y^2}{2\mu_y n}, \quad \mathbb{E}\left\|\mathbf{Y}^{N_y} - \overline{\mathbf{Y}}^{N_y}\right\| \leq \sqrt{\delta_y'}. \tag{31}$$

Remark 7. Note, that the total number of gradient computations and communication steps to approach the qualities (30) and (31) can be considered comparable with (27) and (28).

Remark 8. In the case, when $\sigma = 0$ (i.e., the noise is almost everywhere is zero), Theorem 4.5 gets results worse than Theorem 4.2. It is related to different rounding and inexact estimations in the proof of Theorem 4.5.

5. Numerical experiments

To show the practical performance of the proposed Algorithms 2 and 3, we performed a series of numerical experiments for the Robust Least Squares problem. All experiments were made using Python 3.4, on a computer with Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 1992 Mhz, 4 Core(s), 8 Logical Processor(s), and 8 GB RAM.

Let us consider the following least squares minimization problem

$$\min_{x \in \mathbb{R}^{d_x}} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \|A_i x - y_{i,0}\|^2, \tag{32}$$

for given matrices $A_i \in \mathbb{R}^{d_i \times d_x}$ and vectors $y_{i,0} \in \mathbb{R}^{d_i}$, when $A_i, y_{i,0}$ are placed in n different nodes. In [8], it was proposed a robust version of this problem in the following form

$$\min_{x \in \mathbb{R}^{d_x}} \max_{y \in \mathbb{R}^{d_y}: \|B_i y\| \leq \delta} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \|A_i x - y_{i,0} - B_i y\|^2,$$

where $B_i \in \mathbb{R}^{d_i \times d_y}$, for some $\delta > 0$. The Robust Least Squares problem with soft constraint has the following form

$$\min_{x \in \mathbb{R}^{d_x}} \max_{y \in \mathbb{R}^{d_y}} \left\{ \phi(x,y) = \frac{1}{n} \sum_{i=1}^{n} \phi_i(x,y) \right\}, \tag{33}$$

where

$$\phi_i(x,y) = \frac{1}{2} \|A_i x - y_{i,0} - B_i y\|^2 - \frac{\alpha}{2} \|B_i y\|^2, \tag{34}$$

for some $\alpha > 1$. Note, that the matrices A_i, B_i and the vectors $y_{i,0}$ for each $i \in \{1, \ldots, n\}$, are located in different nodes. So, in the distributed statement, we can not

solve this problem explicitly. At the same time, the global solution has a compact form.

Note that the conditions of Theorem 4.5 are fulfilled for the considered saddle point problem (33) with (34). The function $\phi(\cdot, y)$, for each $y \in \mathbb{R}^{d_y}$, satisfies PL-condition with constant μ_x, and $-\phi(x, \cdot)$, for each $x \in \mathbb{R}^{d_x}$, satisfies PL-condition with constant μ_y, where μ_x and μ_y are the smallest non-zero eigenvalues of the matrices $A = \sum\limits_{i=1}^{n} A_i^\top A_i$ and $B = (\lambda - 1) \sum\limits_{i=1}^{n} B_i^\top B_i$, respectively. Note, that problem (33) is a convex-concave problem but not strongly-convex-strongly-concave until A and B are not full-rank matrices.

Firstly, for the least squares minimization problem (32), we compare the performance of the proposed Algorithm 2 to the recently proposed algorithm DAccGD [33]. In [33], for the least squares problem, it was compared the performance of DAccGD to EXTRA [36], DIGing [26], Mudag [42], and APM-C [20], as a result of the comparison, DAccGD outperformed all the mentioned algorithms.

We run Algorithm 2 and DAccGD, for problem (32), with $d_x = 1000$ and $n = 20$. The matrices A_i and vectors $y_{i,0}$, for each $i \in \{1, \ldots, n\}$, are randomly generated from the standard normal distribution. The graphs $E^k, \forall k \geq 0$ are randomly generated, and the corresponding mixing matrices $\mathbf{W}^k$ are specified according to (10), with Metropolis weights. We take the zero matrices for the initialization. We also choose the fixed step sizes $\gamma_x = 10^{-3}, N_x = 2 \times 10^4, N_y = 100$ and 10 steps in the consensus at each iteration. For the considered problem (32), with the previously mentioned parameters and settings, is ill-conditioned since we have $L \approx 4 \times 10^6$ and $\mu \approx 10^{-9}$, thus the conditions number $\kappa = L/\mu \approx 4 \times 10^{15}$.

The results of the conducted experiments are represented in Figure 1. These results demonstrate the value of the objective function and the norm of its gradient at each generated point x_k by algorithms as a function of iteration k. From Fig. 1, we can see how the proposed Algorithm 2 outperformed DAccGD, and for a not sufficiently big number of iterations (namely 2000) we can achieve a solution to the problem with high accuracy.
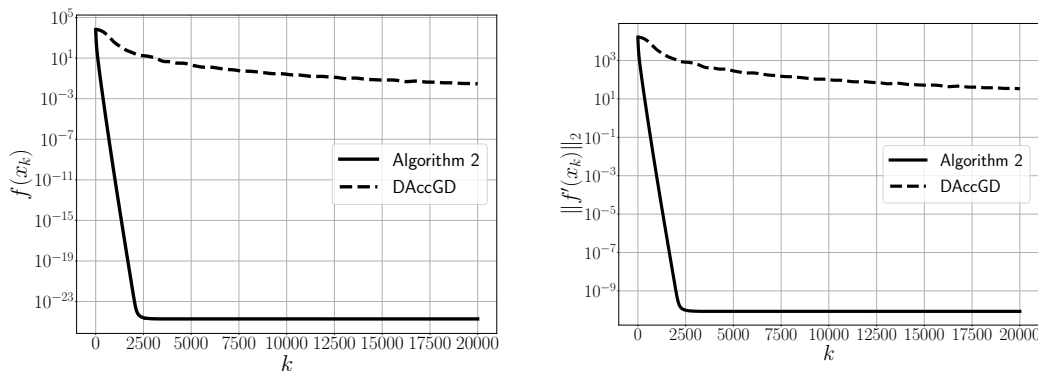

Figure 1.: Results of Algorithms 2 and DAccGD [33], for problem (32).

Also, for problem (33) with (34), we run Algorithm 3 with $d_x = 1000, d_y = 100$ and different values of n (the number of components in (33), which indicates the number of nodes in the graph). We take the zero matrices for the initialization of the algorithm.

The matrices A_i, B_i and vectors $y_{i,0}$, for each $i \in \{1, \ldots, n\}$ in (34) are randomly generated from the standard normal distribution. We also choose the fixed step sizes $\gamma_x = \gamma_y = 10^{-3}, N_x = 10^4, N_y = 10$ (experimentally we see more than 10 iterations, but there is not a remarkable difference) and 10 steps in the consensus at each iteration of Algorithm 3. The results of the conducted experiments, for problem (33) with (34), are represented in Figures 2, 3. These results demonstrate the value of the objective function ϕ and the norm of its gradient with respect to both variables at each point (x_k, y_k) generated by the algorithm as a function of iteration k, and the running time of the algorithm in seconds as a function of the number of nodes n.

From Fig. 2 and Fig. 3, we can see the efficiency of the proposed Algorithm 3, which provides a remarkable quality solution with respect to the value of the objective function and the norm of its gradient at each generated point each iteration. Also, we can see the effect of the number of nodes n, in the work of the algorithm, where the quality of a solution decreases when we increase (not strongly) the number of nodes.
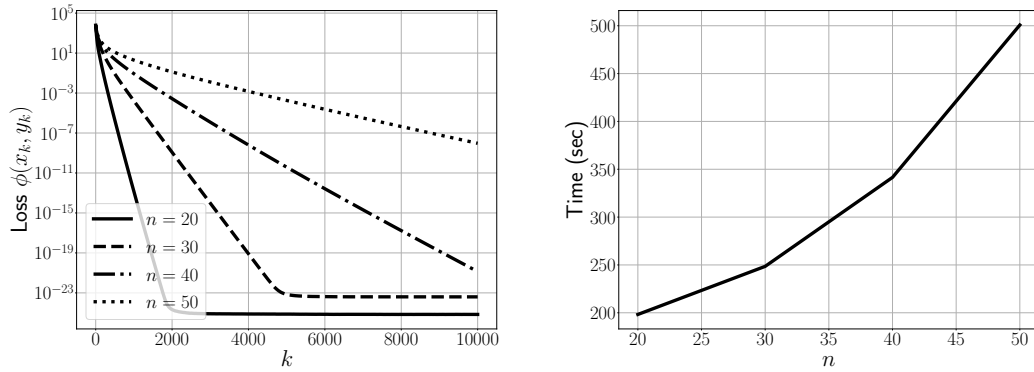

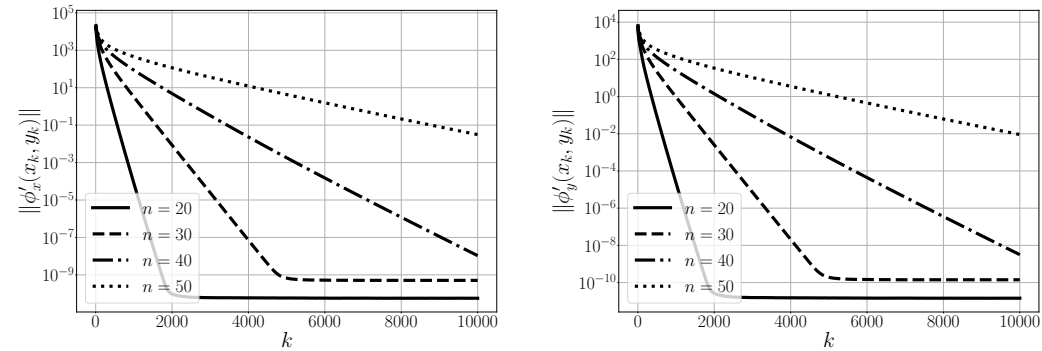Figure 2.: Results of Algorithm 3 for problem (33) with (34), $\alpha = 2$, and different values of n.

Figure 3.: Results of Algorithm 3 for problem (33) with (34), $\alpha = 2$, and different values of n.

6. Conclusion

In this paper, we studied the decentralized minimization problem with objective functions that satisfy Polyak-Łojasiewicz condition (PL-condition), and the decentralized saddle-point problem with a structure of the sum, with objectives satisfying the two-sided PL-condition. For solving the minimization problem under consideration, we proposed a gradient descent type method with a consensus projection procedure and the inexact gradient of the objective function. To solve the considered saddle point problem, we proposed a generalization of the Multi-step Gradient Descent Ascent method with a consensus procedure, and inexact gradients of the objective function concerning both variables. For the studied classes of the problems, we estimated the sufficient communication steps and iterations number to approach the required quality concerning the function and the distance between the agent's vectors and their average some results of the conducted numerical experiments are presented, which demonstrate the effectiveness of the proposed algorithm for the least squares and robust least squares problems.

References

[1] A. Ajalloeian and S.U. Stich, *On the convergence of sgd with biased gradients*, arXiv:2008.00051 (2020).

[2] Y. Arjevani, J. Bruna, B. Can, M. Gurbuzbalaban, S. Jegelka, and H. Lin, *Ideal: Inexact decentralized accelerated augmented lagrangian method*, Advances in Neural Information Processing Systems 33 (2020), pp. 20648–20659.

[3] B. Barazandeh, D.A. Tarzanagh, and G. Michailidis, *Solving a Class of Non-Convex Min-Max Games Using Adaptive Momentum Methods*, in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 3625–3629.

[4] M. Belkin, *Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation*, Acta Numerica 30 (2021), pp. 203–248.

[5] A. Beznosikov, E. Gorbunov, H. Berard, and N. Loizou, *Stochastic gradient descent-ascent: Unified theory and new efficient methods*, arXiv:2202.07262 (2022).

[6] S. Du, J. Lee, H. Li, L. Wang, and X. Zhai, *Gradient descent finds global minima of deep neural networks*, in *International conference on machine learning*. PMLR, 2019, pp. 1675–1685.

[7] D. Dvinskikh and A. Gasnikov, *Decentralized and parallel primal and dual accelerated methods for stochastic convex programming problems*, Journal of Inverse and Ill-posed Problems 29 (2021), pp. 385–405.

[8] L. El Ghaoui and H. Lebret, *Robust solutions to least-squares problems with uncertain data*, SIAM Journal on matrix analysis and applications 18 (1997), pp. 1035–1064.

[9] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi, *Global convergence of policy gradient methods for the linear quadratic regulator*, in *International Conference on Machine Learning*. PMLR, 2018, pp. 1467–1476.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, *Generative adversarial networks*, Communications of the ACM 63 (2020), pp. 139–144.

[11] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A.C. Courville, *Improved training of wasserstein gans*, Advances in neural information processing systems 30 (2017).

[12] H. Hendrikx, F. Bach, and L. Massoulie, *An optimal algorithm for decentralized finite-sum optimization*, SIAM Journal on Optimization 31 (2021), pp. 2753–2783.

[13] A. Ivanova, P. Dvurechensky, A. Gasnikov, and D. Kamzolov, *Composite optimization*

for the resource allocation problem, Optimization Methods and Software 36 (2021), pp. 720–754.

[14] D. Jakovetić, *A unification and generalization of exact distributed first-order methods*, IEEE Transactions on Signal and Information Processing over Networks 5 (2018), pp. 31–46.

[15] H. Karimi, J. Nutini, and M. Schmidt, *Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition*, in *Joint European conference on machine learning and knowledge discovery in databases*. Springer, 2016, pp. 795–811.

[16] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, *A unified theory of decentralized sgd with changing topology and local updates*, in *International Conference on Machine Learning*. PMLR, 2020, pp. 5381–5393.

[17] 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.

[18] A. Kurakin, I.J. Goodfellow, and S. Bengio, *Adversarial Machine Learning at Scale*, in *International Conference on Learning Representations*. 2017. Available at `https://openreview.net/forum?id=BJm4T4Kgx`.

[19] H. Li, C. Fang, W. Yin, and Z. Lin, *A sharp convergence rate analysis for distributed accelerated gradient methods*, arXiv:1810.01053 (2018).

[20] H. Li, C. Fang, W. Yin, and Z. Lin, *Decentralized accelerated gradient methods with increasing penalty parameters*, IEEE transactions on Signal Processing 68 (2020), pp. 4855–4870.

[21] H. Li and Z. Lin, *Revisiting extra for smooth distributed optimization*, SIAM Journal on Optimization 30 (2020), pp. 1795–1821.

[22] H. Li, Z. Lin, and Y. Fang, *Variance reduced extra and diging and their optimal acceleration for strongly convex decentralized optimization*, The Journal of Machine Learning Research 23 (2022), pp. 10057–10097.

[23] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, *Towards deep learning models resistant to adversarial attacks*, arXiv:1706.06083 (2017).

[24] 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)*. IEEE, 2018, pp. 6520–6525.

[25] M.Nouiehed, M. Sanjabi, T. Huang, J. Lee, and M. Razaviyayn, *Solving a class of non-convex min-max games using iterative first order methods*, Advances in Neural Information Processing Systems (2019), p. 14905–14916.

[26] A. Nedic, A. Olshevsky, and W. Shi, *Achieving geometric convergence for distributed optimization over time-varying graphs*, SIAM Journal on Optimization 27 (2017), pp. 2597–2633.

[27] A. Nedic and A. Ozdaglar, *Distributed subgradient methods for multi-agent optimization*, IEEE Transactions on Automatic Control 54 (2009), pp. 48–61.

[28] B. Polyak, *Gradient methods for the minimisation of functionals*, Ussr Computational Mathematics and Mathematical Physics 3 (1963), pp. 864–878.

[29] S. Pu, W. Shi, J. Xu, and A. Nedić, *Push–pull gradient methods for distributed optimization in networks*, IEEE Transactions on Automatic Control 66 (2020), pp. 1–16.

[30] G. Qu and N. Li, *Accelerated distributed nesterov gradient descent*, IEEE Transactions on Automatic Control 65 (2019), pp. 2566–2581.

[31] S. Ram, V. Veeravalli, and A. Nedic, *Distributed non-autonomous power control through distributed convex optimization*, Proceedings - IEEE INFOCOM (2009), pp. 3001 – 3005.

[32] A. Rogozin, M. Bochko, P. Dvurechensky, A. Gasnikov, and V. Lukoshkin, *An accelerated method for decentralized distributed stochastic optimization over time-varying graphs*, in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 3367–3373.

[33] A. Rogozin, V. Lukoshkin, A. Gasnikov, D. Kovalev, and E. Shulgin, *Towards accelerated rates for distributed optimization over time-varying networks*, in *Optimization and Applications: 12th International Conference, OPTIMA 2021, Petrovac, Montenegro, September 27–October 1, 2021, Proceedings 12*. Springer, 2021, pp. 258–272.

[34] K. Scaman, F. Bach, S. Bubeck, Y.T. Lee, and L. Massoulié, *Optimal algorithms for smooth and strongly convex distributed optimization in networks*, in *international conference on machine learning*. PMLR, 2017, pp. 3027–3036.

[35] K. Scaman, F. Bach, S. Bubeck, L. Massoulié, and Y.T. Lee, *Optimal algorithms for non-smooth distributed optimization in networks*, Advances in Neural Information Processing Systems 31 (2018).

[36] W. Shi, Q. Ling, G. Wu, and W. Yin, *Extra: An exact first-order algorithm for decentralized consensus optimization*, SIAM Journal on Optimization 25 (2015), pp. 944–966.

[37] J. Sun, Q. Qu, and J. Wright, *A geometric analysis of phase retrieval*, Foundations of Computational Mathematics 18 (2018), pp. 1131–1198.

[38] C.A. Uribe, S. Lee, A. Gasnikov, and A. Nedić, *A dual approach for optimal algorithms in distributed optimization over networks*, in *2020 Information Theory and Applications Workshop (ITA)*. IEEE, 2020, pp. 1–37.

[39] E. Wei and A. Ozdaglar, *Distributed alternating direction method of multipliers*, in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 5445–5450.

[40] X. Wu and J. Lu, *Fenchel dual gradient methods for distributed convex optimization over time-varying networks*, IEEE Transactions on Automatic Control 64 (2019), pp. 4629–4636.

[41] J. Yang, N. Kiyavash, and N. He, *Global convergence and variance-reduced optimization for a class of nonconvex-nonconcave minimax problems*, arXiv:2002.09621 (2020).

[42] H. Ye, L. Luo, Z. Zhou, and T. Zhang, *Multi-consensus decentralized accelerated gradient descent*, Journal of Machine Learning Research 24 (2023), pp. 1–50.

[43] P. Yue, C. Fang, and Z. Lin, *On the lower bound of minimizing polyak- lojasiewicz functions*, The Thirty Sixth Annual Conference on Learning Theory (2023), pp. 2948–2968.

[44] G. Zhang and R. Heusdens, *Distributed optimization using the primal-dual method of multipliers*, IEEE Transactions on Signal and Information Processing over Networks 4 (2017), pp. 173–187.

Appendix A. Proof of Lemma 3.2.

Proof. Note, for the bias in $\overline{\widetilde{\nabla}}f(\overline{\mathbf{X}}^k)$ we proved in the main part the following estimation

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\overline{\widetilde{\nabla}f}(\mathbf{X}^k) - \nabla f(\overline{x}^k)\right\|^2 \leq \frac{2\delta^2 + L_l^2\delta'}{n}. \tag{A1}$$

Also, we assume that the condition $\mathbb{E}_{\mathbf{X}^j}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\|^2 \leq \delta'$ holds for all $j \leq k$.

Let us estimate noise in $\overline{\widetilde{\nabla}}f(\overline{\mathbf{X}}^k)$. For this, we estimate the second moment of the random component in $\widetilde{\nabla}F(\mathbf{X}^k)$ for given $\overline{\mathbf{X}}^k$. It is given by the expression

$$\mathbb{E}_{\mathbf{X}^k,\xi}\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|.$$

Let us estimate the inner part in the following way:

$$\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\| \leq \left\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k)\right\|$$
$$+ \left\|\mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|$$
$$\leq \mathbb{E}_{\mathbf{X}^k}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|$$
$$+ \left\|\mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|,$$

where we used triangle inequality and Jensen's inequality.

Using these statements again, the second term in the sum above can be represented in the following way:

$$\left\|\mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\| \leq \mathbb{E}_{\mathbf{X}^k}\left\|\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|$$
$$+ \left\|\mathbb{E}_{\mathbf{X}^k}\nabla F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\| + \left\|\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|.$$

From this, we have the following estimation for its square expectation:

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|^2 \leq 2\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k}\nabla F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|^2$$
$$+ 8\mathbb{E}\left\|\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|^2.$$

The term $\left\|\mathbb{E}_{\mathbf{X}^k}\nabla F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|$ can be estimated above by the following sum

$$\mathbb{E}_{\mathbf{X}^k}\left\|\nabla F(\mathbf{X}^k) - \nabla F(\overline{\mathbf{X}}^k)\right\| + \left\|\nabla F(\overline{\mathbf{X}}^k) - \nabla F(\mathbf{X}^k)\right\|.$$

From this we can estimate the full mathematical expectation as $\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k}\nabla F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|^2 \leq 4L_l^2\delta'$ until $\mathbb{E}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta'$. Let us introduce the noise value $n(\mathbf{X}^k) = \widetilde{\nabla}F(\mathbf{X}^k) - \mathbb{E}_{\xi}\widetilde{\nabla}F(\mathbf{X}^k)$ and the bias $b(\mathbf{X}^k) = \mathbb{E}_{\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)$. Note, that $\mathbb{E}\|n(\mathbf{X}^k)\|^2 \leq \sigma^2$ and $\mathbb{E}\left\|b(\mathbf{X}^k)\right\|^2 \leq \delta^2$. So, the expectation of another term can be estimated in the following form:

$$\mathbb{E}\left\|\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\mathbf{X}^k)\right\|^2 = \mathbb{E}\left\|b(\mathbf{X}^k)\right\|^2 + \mathbb{E}\left\|n(\mathbf{X}^k)\right\|^2 \leq \sigma^2 + \delta^2.$$

Finally, we have that

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|^2 \leq 8L_l\sqrt{\delta'} + 8\sigma + 8\delta.$$

Uniting the inequalities above, we have the following estimation:

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|^2 \leq 2\mathbb{E}_{\mathbf{X}^k,\xi}\left(\mathbb{E}_{\mathbf{X}^k}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\right\|\right)^2$$
$$+ 16L_l^2\delta' + 16\sigma^2 + 16\delta^2$$
$$\leq 16L_l^2\delta' + 18\sigma^2 + 16\delta^2.$$

Finally, for the noise component in $\overline{\widetilde{\nabla}} f(\overline{X}^k)$, we have the following estimation:

$$
\begin{aligned}
\mathbb{E}\left\|\mathbb{E}_{X,\xi}\overline{\widetilde{\nabla}} f(X^k) - \overline{\widetilde{\nabla}} f(X^k)\right\|^2 &= \frac{1}{n}\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\overline{\widetilde{\nabla}} F(\mathbf{X}^k) - \overline{\widetilde{\nabla}} F(\mathbf{X}^k)\right\|^2 \\
&\leq \frac{1}{n}\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla} F(\mathbf{X}^k) - \widetilde{\nabla} F(\mathbf{X}^k)\right\|^2 \\
&\leq \frac{16L_l^2\delta' + 18\sigma^2 + 16\delta^2}{n}.
\end{aligned}
$$

So, we obtained the estimation for the noise component in $\overline{\widetilde{\nabla}} f(X^k)$. $\qquad\square$

Appendix B. Proof of Lemma 3.3.

Proof. According to proof of Lemma 2 from [1] we can prove the following inequality:

$$
\mathbb{E}\left[f(x^{t+1})\Big|x^t\right] \leq f(x^t) - \frac{\gamma}{2}\left\|\nabla f(x^t)\right\|^2 + \frac{\gamma}{2}\left\|b(x^k)\right\|^2 + \frac{\gamma^2 L}{2}\mathbb{E}_\xi\left\|n(x^k)\right\|^2.
$$

Using PL-condition for the function f and taking full mathematical expectation, we have the following inequality

$$
\mathbb{E}\left[f(x^{t+1}) - f^*\right] \leq (1 - \gamma\mu)\mathbb{E}\left[f(x^t) - f^*\right] + \frac{\gamma}{2}\mathbb{E}\left\|b(x^k)\right\|^2 + \frac{\gamma^2 L}{2}\mathbb{E}\|n(x^k)\|^2.
$$

Using lemma's conditions we obtain the following inequality:

$$
\mathbb{E}\left[f(x^{t+1}) - f^*\right] \leq (1 - \gamma\mu)\mathbb{E}\left[f(x^t) - f^*\right] + \frac{\gamma\delta^2}{2} + \frac{\gamma^2 L\sigma^2}{2},
$$

or

$$
\mathbb{E}\left[f(x^k) - f^*\right] \leq (1 - \gamma\mu)^k\left(f(x^0) - f^*\right) + \frac{\delta^2 + \gamma L\sigma^2}{2\mu}.
$$

$\qquad\square$

Appendix C. Proof of Theorem 3.4.

Proof. Using Assumptions 2.2, Lemma 3.3, taking step size $\gamma = \frac{1}{L_g}$, we can estimate convergence for function f:

$$
\mathbb{E}\left[f(\overline{x}^k) - f^*\right] \leq \left(1 - \frac{\mu}{L_g}\right)^k\left(f(\overline{x}^0) - f^*\right) + \frac{\Delta^2}{2\mu n}, \tag{C1}
$$

until $\mathbb{E}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\| \leq \delta'$ for $j < k$ where $\Delta^2 = 18\left(L_l^2\delta' + \sigma^2 + \delta^2\right)$.

But really on iterations of Algorithm 2 we have access only to $\mathbf{X}^k$. Let us find the sufficient communication step such that the following expression is true:

$$\mathbb{E}\left\|\mathbf{X}^k - \overline{\mathbf{X}}^k\right\|^2 \leq \delta' \implies \mathbb{E}\left\|\mathbf{X}^{k+1} - \overline{\mathbf{X}}^{k+1}\right\|^2 \leq \delta'. \tag{C2}$$

By contraction property, we have

$$\left\|\mathbf{X}^{k+1} - \overline{\mathbf{X}}^{k+1}\right\| \leq (1-\lambda)^{\left\lfloor \frac{T_k}{\tau} \right\rfloor}\left\|\mathbf{Z}^{k+1} - \overline{\mathbf{X}}^{k+1}\right\|. \tag{C3}$$

Here we used, that $\overline{\mathbf{Z}}^{k+1} = \overline{\mathbf{X}}^{k+1}$. Let us estimate the right part of (C3).

$$
\begin{aligned}
\mathbb{E}\left\|\overline{\mathbf{X}}^{k+1} - \mathbf{Z}^{k+1}\right\|^2 &\leq 2\mathbb{E}\left\|\overline{\mathbf{X}}^k - \mathbf{X}^k\right\|^2 + 2\gamma^2\mathbb{E}\left\|\widetilde{\nabla}F(\mathbf{X}^k)\right\|^2 \\
&\leq 2\delta' + 6\gamma^2\mathbb{E}\left\|\widetilde{\nabla}F(\mathbf{X}^k) - \nabla F(\overline{\mathbf{X}}^k)\right\|^2 \\
&\quad + 6\gamma^2\mathbb{E}\left\|\nabla F(\overline{\mathbf{X}}^k) - \nabla F(\overline{\mathbf{X}}^*)\right\|^2 + 6\gamma^2\left\|\nabla F(\overline{\mathbf{X}}^*)\right\|^2 \\
&\leq 2\delta' + 6\gamma^2\Delta^2 + 6\gamma^2 L_g^2\mathbb{E}\left\|\overline{\mathbf{X}}^k - \overline{\mathbf{X}}^*\right\|^2 + 6\gamma^2\left\|\nabla F(\overline{\mathbf{X}}^*)\right\|^2.
\end{aligned}
$$

From quadratic growth condition (9) we have

$$\mathbb{E}\left\|\overline{\mathbf{X}}^k - \overline{\mathbf{X}}^*\right\|^2 = n\mathbb{E}\left\|\overline{x}^k - \overline{x}^*\right\|^2 \leq \frac{2n}{\mu}\mathbb{E}\left[f(x^k) - f^*\right].$$

Further, using the convergence rate (19) we can estimate the third value in the sum above:

$$\mathbb{E}\left\|\overline{\mathbf{X}}^k - \overline{\mathbf{X}}^*\right\|^2 \leq \frac{2n}{\mu}\left(1 - \frac{\mu}{L_g}\right)^{k+1}\left(F(\overline{\mathbf{X}}^0) - F^*\right) + \frac{\Delta^2}{\mu^2}.$$

Note, that $\mathbb{E}\xi \leq \sqrt{\mathbb{E}\xi^2}$ for any random value ξ. Finally, we have that:

$$\mathbb{E}\left\|\overline{\mathbf{X}}^{k+1} - \mathbf{Z}^{k+1}\right\|^2 \leq D,$$

for D such that:

$$D = 6\gamma^2\|\nabla F(\overline{\mathbf{X}}^*)\|^2 + 2\delta' + 6\left(\gamma^2 + \frac{1}{\mu^2}\right)\Delta^2 + \frac{12\gamma^2 L_g^2}{\mu}\left(1 - \frac{\mu}{L_g}\right)\left(F(\overline{\mathbf{X}}^0) - F^*\right).$$

So, for $T_k = \tau\left\lceil \frac{1}{2\lambda}\log\left(\frac{D}{\delta'}\right)\right\rceil$ the condition (C2) is met. $\qquad\square$

Appendix D. Proof of Lemma 4.3.

Proof For the given $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$, we can estimate the bias in inexact gradient with respect

to Y in the following form:

$$
\begin{aligned}
\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla_Y\Phi(\overline{\mathbf{X}},\overline{\mathbf{Y}})\right\|^2 \leq & 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \mathbb{E}_{\mathbf{X}^k,\mathbf{Y}}\nabla_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2 \\
& + 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k,\mathbf{Y}}\nabla_Y\Phi(\mathbf{X},\mathbf{Y}) - \mathbb{E}_{\mathbf{X}^k}\nabla_{\mathbf{Y}}\Phi(\mathbf{X},\overline{\mathbf{Y}})\right\|^2 \\
& + 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}^k}\nabla_{\mathbf{Y}}\Phi(\mathbf{X},\overline{\mathbf{Y}}) - \nabla_{\mathbf{Y}}\Phi(\overline{\mathbf{X}},\overline{\mathbf{Y}})\right\|^2 \\
\leq & 3\mathbb{E}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}_Y\Phi(\mathbf{X},\mathbf{Y}) - \nabla_Y\Phi(\mathbf{X},\mathbf{Y})\right\|^2 \\
& + 3\mathbb{E}\|\nabla_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla_Y\Phi(\mathbf{X},\overline{\mathbf{Y}})\|^2 \\
& + 3\mathbb{E}\left\|\nabla_Y\Phi(\mathbf{Y},\overline{\mathbf{Y}}) - \nabla_{\mathbf{Y}}\Phi(\overline{\mathbf{X}},\overline{\mathbf{Y}})\right\|^2 \\
\leq & 3\delta^2 + 3L_{yy}^2\delta_y' + 3L_{yx}^2\delta_x'.
\end{aligned}
$$

In the inequality above we used fact that $\|a + b + c\|^2 \leq 3\|a\|^2 + 3\|b\|^2 + 3\|c\|^2$ for each vectors a, b and c. Note, for the bias in $\overline{\widetilde{\nabla}}f(\overline{X}^k)$ we proved in the main part the following estimation

$$
\mathbb{E}\left\|\mathbb{E}_{X,\xi}\overline{\widetilde{\nabla}f}(X^k) - \nabla f(\overline{x}^k)\right\|^2 \leq \frac{2\delta^2 + L_l^2\delta'}{n}. \tag{D1}
$$

Also, we assume that the condition $\mathbb{E}_{\mathbf{X}^k}\left\|\mathbf{X}^j - \overline{\mathbf{X}}^j\right\|^2 \leq \delta'$ holds for all $j \leq k$.

Let us estimate noise in $\overline{\widetilde{\nabla}}f(\overline{X}^k)$. For this, we estimate the second moment of the random component in $\widetilde{\nabla}F(\mathbf{X}^k)$ for given $\overline{\mathbf{X}}^k$. It is given by the expression $\mathbb{E}_{\mathbf{X}^k,\xi}\|\mathbb{E}_{\mathbf{X}^k,\xi}\widetilde{\nabla}F(\mathbf{X}^k) - \widetilde{\nabla}F(\mathbf{X}^k)\|$. Let us estimate the inner part in the following way:

$$
\begin{aligned}
\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_Y\Phi(\mathbf{X},\mathbf{Y})\right\| \leq & \left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \mathbb{E}_{\mathbf{X},\mathbf{Y}}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\| \\
& + \left\|\mathbb{E}_{\mathbf{X},\mathbf{Y}}\widetilde{\nabla}_Y\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_Y\Phi(\mathbf{X},\mathbf{Y})\right\| \\
\leq & \mathbb{E}_{\mathbf{X},\mathbf{Y}}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}_Y\Phi(X,Y) - \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\| \\
& + \left\|\mathbb{E}_{\mathbf{X},\mathbf{Y}}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\|.
\end{aligned}
$$

The last term can be estimated in a similar way as in Appendix A:

$$
\begin{aligned}
\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y}}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2 \leq & 2\mathbb{E}\|\mathbb{E}_{\mathbf{X},\mathbf{Y}}\nabla_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\|^2 \\
& + 8\sigma^2 + 8\delta^2 \\
\leq & 8L_{yy,l}^2\delta_y' + 8L_{yx,l}^2\delta_x' + 8\sigma^2 + 8\delta^2,
\end{aligned}
$$

Finally, we can estimate required the second moment of noise component:

$$
\begin{aligned}
\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2 \leq & 2\mathbb{E}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}_Y\Phi(\mathbf{X},\mathbf{Y}) - \widetilde{\nabla}_{\mathbf{Y}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2 \\
& + 16L_{yy,l}^2\delta_y' + 16L_{yx,l}^2\delta_x' + 16\sigma^2 + 16\delta^2 \\
\leq & 16L_{yy,l}^2\delta_y' + 16L_{yx,l}^2\delta_x' + 18\sigma^2 + 16\delta^2.
\end{aligned}
$$

27

Thus we can conclude, that

$$\mathbb{E}\left\|\mathbb{E}\overline{\widetilde{\nabla}_{\mathbf{Y}}\phi}(\mathbf{X},\mathbf{Y}) - \overline{\widetilde{\nabla}_{\mathbf{Y}}\phi}(\mathbf{X},\mathbf{Y})\right\| \leq \frac{16L_{yy,l}^2\delta_y' + 16L_{yx,l}^2\delta_x' + 18\sigma^2 + 16\delta^2}{n}. \qquad \text{(D2)}$$

$\square$

Appendix E. Proof of Lemma 4.4

Proof. In a similar way, like for the gradient with respect to the inner variable $\mathbf{Y}$, we can estimate the mathematical expectation of bias with respect to the outer variable $\mathbf{X}$:

$$\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla F\left(\overline{\mathbf{X}}\right)\right\|^2 \leq 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},\mathbf{Y},\xi}\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}) - \mathbb{E}_{\mathbf{X},\mathbf{Y}}\nabla_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2$$

$$+ 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X},Y}\nabla_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}) - \mathbb{E}_{\mathbf{X}}\nabla_{\mathbf{X}}\Phi(\mathbf{X},\overline{\mathbf{Y}}^*(\mathbf{X}))\right\|^2$$

$$+ 3\mathbb{E}\left\|\mathbb{E}_{\mathbf{X}}\nabla_{\mathbf{X}}\Phi(\mathbf{X},\overline{\mathbf{Y}}^*(\mathbf{X})) - \nabla_{\mathbf{X}}\Phi(\overline{\mathbf{X}},\overline{\mathbf{Y}}^*(\mathbf{X}))\right\|^2$$

$$\leq 3\mathbb{E}\left\|\mathbb{E}_{\xi}\widetilde{\nabla}_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y})\right\|^2$$

$$+ 3\mathbb{E}\left\|\nabla_{\mathbf{X}}\Phi(\mathbf{X},\mathbf{Y}) - \nabla_{\mathbf{X}}\Phi(\mathbf{X},\overline{\mathbf{Y}}^*(\mathbf{X}))\right\|^2$$

$$+ 3\mathbb{E}\left\|\nabla_{\mathbf{X}}\Phi(\mathbf{X},\overline{\mathbf{Y}}^*(\mathbf{X}) - \nabla_{\mathbf{X}}\Phi(\overline{\mathbf{X}},\overline{\mathbf{Y}}^*(\mathbf{X}))\right\|^2$$

$$\leq 3\delta^2 + 3L_{xx}^2\delta_x' + 6L_{xy}^2\left(\frac{2\varepsilon_y}{\mu_y} + \frac{\Delta_y^2}{\mu_y^2 n} + \delta_y'\right).$$

Note, that the noise component in the inexact gradient $\overline{\widetilde{\nabla}_{\mathbf{X}}\phi}(\mathbf{X},\mathbf{Y})$ does not depend on inexactness on a solution by the inner problem. So, repeating the steps above for inexact gradient with respect to the variable $\mathbf{X}$, we can obtain the following estimation

$$\mathbb{E}\left\|\mathbb{E}\overline{\widetilde{\nabla}_{\mathbf{X}}\phi}(\mathbf{X},\mathbf{Y}) - \overline{\widetilde{\nabla}_{\mathbf{X}}\phi}(\mathbf{X},\mathbf{Y})\right\| \leq \frac{16L_{xy,l}^2\delta_y' + 16L_{xx,l}^2\delta_x' + 18\sigma^2 + 16\delta^2}{n}. \qquad \text{(E1)}$$

Note, that the variable Y in (E1) is a result of the inner loop in Algorithm 3. The estimations (D2) and (E1) give the bounds on the second norm of noise. $\square$