Accelerated Methods with Compression for Horizontal and Vertical Federated Learning

Anonymous Authors[1]

Abstract

Distributed optimization algorithms have emerged as a superior approaches for solving machine learning problems. To accommodate the diverse ways in which data can be stored across devices, these methods must be adaptable to a wide range of situations. As a result, two orthogonal regimes of distributed algorithms are distinguished: horizontal and vertical. During parallel training, communication between nodes can become a critical bottleneck, particularly for high-dimensional and over-parameterized models. Therefore, it is crucial to enhance current methods with strategies that minimize the amount of data transmitted during training while still achieving a model of similar quality. This paper introduces two accelerated algorithms with various compressors, working in the regime of horizontal and vertical data division. By utilizing a momentum and variance reduction technique from the `Katyusha` algorithm, we were able to achieve acceleration and demonstrate one of the best asymptotics for the horizontal case. Additionally, we provide one of the first theoretical convergence guarantees for the vertical regime. Our experiments involved several compressor operators, including RandK and PermK, and we were able to demonstrate superior practical performance compared to other popular approaches.

1. Introduction

As machine learning continues to gain popularity, it is essential for training algorithms to maintain exceptionally high performance in order to effectively address real-world issues. To increase the quality of models' predictions, larger datasets are used. However, training models on a huge amount of samples on a single machine can be very time-consuming, which is why faster optimization approaches are needed.

To address this problem, the scientific community came to distributed algorithms, where the calculation process is divided among different devices (also called machines, nodes, clusters, or workers). Such parallel computation can be used in situations, where data is distributed across several machines, as in the cases of classical distributed (Verbraeken et al., 2020) and federated learning approaches (Konečný et al., 2017). The latter can be divided into two different regimes. One of these is horizontal federated learning, which is the most popular and extensively studied approach. In this scenario, each worker possesses their own collection of samples, but share the same set of features. A different situation is considered in the vertical case, (Zhang et al., 2021), where every device has a unique set of features of the same samples. The practical use of the latter regime can be seen in a simple example, where a digital finance company wants to evaluate the risk of approving a loan. To make a qualified prediction, it needs to collect different types of data (features) from the same person (sample). For example, online shopping information from an E-commerce company or average monthly deposit and account information from a bank. However, legal restrictions or competition between participants prevent these organizations from sharing data, which is why a vertical case is suitable in this situation (Gu et al., 2020). But classical distributed and horizontal federative learning methods can not be forgotten, as they can show a great performance in accelerating training speed (Goyal et al., 2018). This is why distributed optimization is gaining a lot of attention nowadays.

However, parallelization of a task practically does not lead to an ideal decrease in time. That means that having N devices does not accelerate task by N times. This happens because of limited ability of networks to exchange information. Thus, the key bottleneck of parallel computation is the communication part. There have been considered several ways of dealing with this issue (Konečnỳ et al., 2016; Smith et al., 2018), but in our paper we concentrate solely on re-

---
[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

ducing communication cost of each iteration by decreasing the size of sending information also known as compression technique (Chilimbi et al., 2014; Alistarh et al., 2017). Due to reducing quality of communicated information, gradient descent methods perform greater amount of iterations, but overall time complexity can be reduced.

Compression mathematically can be represented in the form of a vector function, which is called compression operator. It takes a vector and returns a vector of the same dimension, but the last usually takes less resources in communication networks. In our paper, we consider such operators with the following properties:

Definition 1.1 *We say that the compression operator Q is unbiased, if*

$$\mathbb{E}\left[Q(x)\right] = x \, , \, \forall x \in \mathbb{R}^d.$$

We also assume that there exists a constant $w \geq 1$, such:

$$\mathbb{E}\left[\|Q(x)\|^2\right] \leq \omega \|x\|^2 \, , \, \forall x \in \mathbb{R}^d.$$

We say that the compressor operator Q has density coefficient β, if $Q(x)$ requires in β times less space complexity than x.

1.1. Related works

Unbiased compression. An idea to shrink the vector was researched in (Nesterov, 2012). This work developed a variant of single-node gradient descent in which only some random coordinates are updated at every step. Later, in the literature on compression, operators of this kind began to be called "RandK" (Beznosikov et al., 2024). In the paper by (Richtárik and Takáč, 2013), coordinate descent method was adapted for distributed optimization. Generalization with arbitrary unbiased compressor was firstly introduced in QSGD paper (Alistarh et al., 2017), but this method does not converge to the true solution, but rather to some neighbourhood, dependent on functions' variance (Gorbunov, 2021).

Variance reduction. The problem mentioned above is characterized by a non-zero limit in the optimal point of the variance of stochastic gradient. A technique of choosing a vector of step direction to fix this challenge is called variance reduction. Firstly used in standard optimization problem (Johnson and Zhang, 2013; Nguyen et al., 2017) and later was adapted for distributed optimization in DIANA (Mishchenko et al., 2023), by compressing not gradient itself, but rather gradient difference, and later for non-convex distributed case in MARINA (Gorbunov et al., 2022). This technique is also used in variational inequalities, for example, in (Beznosikov et al., 2023).

Acceleration. The optimal algorithm for a non-distributed strongly convex problem, which uses variance reduction technique, is Katyusha (Allen-Zhu, 2016). It is based on "negative momentum" idea with the combination of stochastic adjustment of the gradient in the old point. Vector in which gradient is calculated, updates with the loop, which is slower than a coin flip update represented in L-Katyusha (Kovalev et al., 2019).

1.2. Our contributions

• **New distributed algorithms.** We present new distributed algorithms with compression for various regimes of data division. As mentioned before, parallel algorithms can be based on non-parallel approaches with variance reduction, e.g., for non-convex problem MARINA (Gorbunov et al., 2022) is based on PAGE (Li et al., 2021), or for variational inequalities MASHA (Beznosikov et al., 2023) is based on (Alacaoglu and Malitsky, 2022). In our paper, we base our results on L-Katyusha (Kovalev et al., 2019), one of the state-of-the-art algorithm for strongly convex problem with variance reduction technique.

• **Horizontal regime.** We propose L-Katyusha-based algorithm with horizontal data division – DHPL-Katyusha (Algorithm 1), in which every worker compresses and sends the difference between gradients of his function in new and old points, after which regular L-Katyusha algorithm is performed locally. Our algorithm has asymptotics in comparison with other popular approaches (Table 1). Unlike QSGD (Alistarh et al., 2017), DIANA (Mishchenko et al., 2023), VR-DIANA (Horváth et al., 2019), MASHA (Beznosikov et al., 2023) and Three Pillars Algorithm (Beznosikov and Gasnikov, 2023) our algorithm is accelerated (e.g. has asymptotics proportional to $\sqrt{\frac{L}{\mu}}$), and in comparison with ADIANA, DHPL-Katyusha has the same asymptotics in the best regime $\omega = n$ but shows better empirical results in experiments.

• **Vertical regime.** We strive to present DVPL-Katyusha (Algorithm 2) — the algorithm with compression, that works under the assumption of the vertical data division with the MSE loss. Every worker sends its own part of scalar values, needed to calculate total loss, and then perform other part of L-Katyusha locally with its own set of features. We estimate the asymptotics in such setting and show a lower bound of it on a constructive example. We are one of the first to present algorithm with vertical regime and guaranties for theoretical convergence. Other approaches, represented in Table 1, such as AVFL (Cai et al., 2022), CE-VFL (Sun et al., 2023), SecureBoost+ (Chen et al., 2021) and eHE-SecureBoost (Xu et al., 2021) do not have it. CVFL algorithm (Castiglia et al., 2023) not just has theoretical convergences rate only for special compressors, but also does not have concrete convergence guaranties. The CVFL algorithm (Castiglia et al., 2023) has theoretical convergence rates only for special compressors, but also

Table 1: Summary of bounds for iteration complexities for finding an ε-solution. Convergence is measured by the distance to the solution.

Regime	Reference	Iteration complexity
Horizontal	QSGD (Alistarh et al., 2017) [1]	$\mathcal{O}\left(\frac{L}{\mu}\left(1+\frac{w-1}{n}\right)\log\frac{1}{\varepsilon}\right)$
	DIANA (Mishchenko et al., 2023)	$\mathcal{O}\left(\left(\frac{L}{\mu}\left(1+\frac{2(\omega-1)}{n}\right)+\omega\right)\log\frac{1}{\varepsilon}\right)$
	VR-DIANA (Horváth et al., 2019)	$\mathcal{O}\left(\left(\frac{L}{\mu}\left(1+\frac{\omega-1}{n}\right)+\omega\right)\log\frac{1}{\varepsilon}\right)$
	ADIANA (Li et al., 2020)	$\mathcal{O}\left(\left(\sqrt{\frac{L}{\mu}}\left(1+\sqrt{\left(\frac{\omega-1}{n}+\sqrt{\frac{\omega-1}{n}}\right)\omega}\right)+\omega\right)\log\frac{1}{\varepsilon}\right)$
	MARINA (Gorbunov et al., 2022) [2]	$\mathcal{O}\left(\left(\frac{L}{\mu}\left(1+\frac{\omega-1}{\sqrt{n}}\right)+w\right)\log\frac{1}{\varepsilon}\right)$
	MASHA (Beznosikov et al., 2023) [3]	$\mathcal{O}\left(\left(\frac{L}{\mu}\sqrt{\left(w+\frac{(\omega-1)^2}{n}\right)}+\omega\right)\log\frac{1}{\varepsilon}\right)$
	Three Pillars Algorithm (Beznosikov and Gasnikov, 2023) [3, 4]	$\mathcal{O}\left(\left(\frac{L}{\mu}\sqrt{n}+n\right)\log\frac{1}{\varepsilon}\right)$
	Algorithm 1(This paper)	$\mathcal{O}\left(\left(\sqrt{\frac{L}{\mu}}\sqrt{\omega\left(\frac{\omega-1}{n}+1\right)}+\omega\right)\log\frac{1}{\varepsilon}\right)$
Vertical	AVFL (Cai et al., 2022); CE-VFL (Sun et al., 2023); SecureBoost+ (Chen et al., 2021); eHE-SecureBoost (Xu et al., 2021);	No theoretical results
	CVFL (Castiglia et al., 2023)	No concrete number of iterations [5]
	Algorithm 2(This paper)	$\mathcal{O}\left(s\left(\sqrt{\frac{L}{\mu}}\sqrt{\omega\left(\frac{1}{s}+(w-1)\frac{sL^2}{\mu^2}\right)}+\omega\right)\log\frac{1}{\varepsilon}\right)$

[1] correct tuning of step size, no convergence with fixed step size; [2] under PŁ condition; [3] for VI and SPPs; [4] for PermK compressor; [5] for special compressor convergence rate is $O(\frac{1}{\sqrt{K}})$, no guaranties in the case of arbitrary compressor;

Notation: μ = constant of strong convexity, L = smoothness constant of the target function, ω = compression constant (see Definition 1.1), n = number of workers, s = number of samples in the MSE problem, $L^2 = \frac{1}{s}\sum_{j=1}^{s} L_j^2$, where L_j is a smoothness constant of a loss on a single sample.

lacks concrete convergence guarantees.

• **Various compressors.** We obtain similar estimates for different compressor. By obtaining ω from Definition 1.1, we research RandK operator, which retains only K random coordinates from a vector. In this paper, we also implement one of the state-of-the-art compressor PermK (Szlendak et al., 2021), which distributes random permutations of vector components across all workers to compute only on them. As PermK can be represented in the form of several correlated compressors, ω can not be defined for it. Thus, a slightly different analysis is used in order to develop a proof.

• **Numerical experiments.** In numerical experiments, we illustrate the most important properties of the new methods. The results correspond to the theory developed.

1.3. Technical preliminaries
Notations. $\langle x, y\rangle := \sum_{i=1}^{d} x_i y_i$ is used to denote standard inner product of $x, y \in \mathbb{R}^d$. Operator $\mathbb{E}[\cdot]$ denotes mathematical expectation. We denote $A_i^T \in \mathbb{R}^d$ as the i-th column of the matrix A^T.

2. Horizontal Case
In this section, we provide an algorithm for the horizontal division of data, which we call DHPL-Katyusha (Algorithm 1). We state the standard distributed learning prob-

lem: each worker has its own dataset, on which it can calculate loss. The target loss function, which we need to minimize, is obtained by summarizing and averaging all workers' losses. Formally, it can be written in the form of:

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

Communications between nodes can be centralized, e.g. they can be managed only by the server, which in gradient approaches collects functions' gradients, averages them, does subsequent calculation to find a new point and broadcasts it to other devices. Such setting is considered to be popular, especially in theory . Another way of organizing communications is a decentralized setup, where all devices are connected in a single network with specific topology and, unlike centralized, can communicate with each other through networks' edges.

The situation in which each worker's compressor is completely independent of the compressor operators of other devices seems to be critical for the decentralised setup . There are several reasons for such difficulties arising, the first is that with such approach we are not able to exploit data similarity between devices to the fullest extent. The second problem we face lies on the level of technical implementation of communications in a network, the size of

the transmitted packets in which is unknown in advance. Thus, it becomes complicated to perform such important operations as AllReduce (Ernie Chan1 and van de Geijn1, 2006) using arbitrary compressors. This is why correlated compressors, like PermK (Szlendak et al., 2021), are used in communication networks. In our paper we research both correlated and uncorrelated compressors, therefore to make algorithm suitable for different ways of organizing communications we change the name of specific communication operation with the word "broadcast".

2.1. Introduction of DHPL-Katyusha

At the beginning of the k-th iteration of DHPL-Katyusha, every worker computes x^k locally, as a convex combination of z^k, w^k and y^k and broadcasts it to all other workers (line 3). After this, each worker already has the same old and new point and therefore can compute the difference between local gradients in the current point x^k and in the old point w^k in the same manner (line 4). Then, each worker applies local compressor and sends this difference to everyone else (line 5, line 6). Having done that, every device is able to find the average of all compressed differences and add full gradient in w^k (stored locally on each device) to apply the variance reduction technique (line 7). After that, "negative momentum" idea is used from (Kovalev et al., 2019) (line 8, line 9). Finally, w^k is updated to the actual point, based on a coin flip, which all workers do with the same random seed (line 10). If w^k is adjusted, the uncompressed gradient is calculated by AllReduce procedure (line 13). As p strives to zero with $\omega \to \infty$, full gradient is updated rarely.

Note that the Q_i compressors in this algorithm work independently of each other.

2.2. Convergence results

To prove the asymptotics of DHPL-Katyusha we need to make the following assumptions for the problem (1).

Assumption 2.1 *Functions $f_i : \mathbb{R}^d \to \mathbb{R}$ are L-smooth for some $L > 0, \forall i \in \overline{1, n}$:*

$$f_i(y) \leq f_i(x) + \langle \nabla f_i(x), y - x \rangle + \frac{L}{2} \|y - x\|^2, \ \forall x, y \in \mathbb{R}^d.$$

Note that with such assumption, the target function $f(x)$ is also L - smooth.

Assumption 2.2 *The function $f : \mathbb{R}^d \to \mathbb{R}$ is μ - strongly convex for some $\mu > 0$:*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2, \ \forall x, y \in \mathbb{R}^d.$$

In the original paper (Kovalev et al., 2019), one of the steps in the proof of the asymptotics of L-Katyusha, which is called as Lemma 6, is the estimation of a term $\|g^k - \nabla f(x^k)\|^2$. It can be formally written in the follow-

---

Algorithm 1 DHPL-Katyusha

Input: initial $y^0 = \omega^0 = z^0 \in \mathbb{R}^d$, step size $\eta = \frac{\theta_2}{(1+\theta_2)\theta_1}$, $\sigma = \frac{\mu}{\widetilde{L}}$, where $\widetilde{L} = L \cdot \max\left\{1, \frac{\omega}{n}\right\}$, parameters $\theta_1, \theta_2 \in R$ and probability $p \in (0, 1]$, every worker has the same random seed.

1: **for** $k = 0, 1, 2, \dots K$ **do**
2:     **for** $i = 1 \dots n$ in parallel **do**
3:         $x^k \leftarrow \theta_1 z^k + \theta_2 \omega^k + (1 - \theta_1 - \theta_2) y^k$
4:         $g_i^k \leftarrow \nabla f_i(x^k) - \nabla f_i(\omega^k)$
5:         $\widetilde{g}_i^k \leftarrow Q_i(g_i^k)$
6:         Using communications broadcast $\widetilde{g}_i^k$
7:         Compute $\widetilde{g}^k \leftarrow \frac{1}{n} \sum_{i=1}^n \widetilde{g}_i^k + \nabla f(\omega^k)$
8:         $z^{k+1} \leftarrow \frac{1}{1+\eta\sigma}(\eta\sigma x^k + z^k - \frac{\eta}{\widetilde{L}}\widetilde{g}^k)$
9:         $y^{k+1} \leftarrow x^k + \theta_1(z^{k+1} - z^k)$
10:       $\omega^{k+1} \leftarrow \begin{cases} y^k, & \text{with probability } p \\ \omega^k, & \text{with probability } 1 - p \end{cases}$
11:         **if** $w^{k+1} = y^k$ **then**
12:             Using communications broadcast $\nabla f_i(w^{k+1})$
13:             Compute $\nabla f(w^{k+1}) = \frac{1}{n} \sum_{i=1}^n \nabla f_i(w^{k+1})$
14:         **end if**
15:     **end for**
16: **end for**

---

ing form:

Lemma 2.3 *For the original L-Katyusha algorithm, the norm of the difference between the true and the real gradient can be estimated as*

$$\|g^k - \nabla f(x^k)\|^2$$
$$\leq 2L\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k), \omega^k - x^k \rangle\right).$$

However, in our paper, g^k is changed to $\widetilde{g}^k$, and therefore, it takes to utilize additional inequalities to gain similar terms for an upper estimate of $\|\widetilde{g}^k - \nabla f(x^k)\|^2$.

Lemma 2.4 *For the DHPL-Katyusha algorithm, the norm of the difference between the true and the real gradient can be estimated as*

$$\|\widetilde{g}^k - \nabla f(x^k)\|^2$$
$$\leq 2\widetilde{L}\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k), \omega^k - x^k \rangle\right),$$

where

$$\widetilde{L} = L\left(\frac{\omega}{n} + 1\right).$$

Note that since $\widetilde{L} \geq L$, all smoothness-related inequalities that originally were true for L would also work for $\widetilde{L}$, so we can prove estimates for `DHPL-Katyusha` using similar steps from (Kovalev et al., 2019), by changing the Lipschitz constant. Therefore, we come to an important definition.

Definition 2.5 *In Katyusha-based algorithm we call $\widetilde{L}$ the efficient Lipschitz constant, if it is the smallest constant, that is greater than L, such:*

$$\|\widetilde{g}^k - \nabla f(x^k)\|^2$$
$$\leq 2\widetilde{L}\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k), \omega^k - x^k \rangle\right).$$

Using the above idea, we can formulate a theorem of convergence:

Theorem 2.6 *Let Assumptions 2.1, 2.2 be hold. Denote $\widetilde{L}$ as $L\left(\frac{\omega}{n} + 1\right)$ and x^* as the solution for the problem 1. Then after k iterations of `DHPL-Katyusha`*

$$\mathbb{E}\left[\mathbb{Z}^{k+1} + \mathbb{Y}^{k+1} + \mathbb{W}^{k+1}\right]$$
$$\leq \frac{1}{1+\eta\sigma}\mathbb{Z}^k + (1 - \theta_1(1-\theta_2))\mathbb{Y}^k + \left(1 - \frac{p\theta_1}{1+\theta_1}\right)\mathbb{W}^k,$$

where:

$$\mathbb{Z}^k := \frac{\widetilde{L}(1+\eta\sigma)}{2\eta}\|z^k - x^*\|^2,$$
$$\mathbb{Y}^k := \frac{1}{\theta_1}\left(f(y^k) - f(x^*)\right),$$
$$\mathbb{W}^k := \frac{\theta_2(1+\theta_1)}{p\theta_1}\left(f(\omega^k) - f(x^*)\right).$$

We choose the concrete p with the aim of reducing the average amount of information sent in each communication procedure. At each iteration, on average, workers communicate at a cost of $O\left(\frac{1}{\beta} + p\right)$. Therefore, p should be equal to $\frac{1}{\beta}$ to get a gain in asymptotics. Other parameters we choose are similar to that of `L-Katyusha` (Kovalev et al., 2019).

Corollary 2.7 *Denote β as a density coefficient of compressor operator Q from Definition 1.1. Let $p = \frac{1}{\beta}$, $\theta_1 = \min\{\sqrt{\frac{2\sigma\beta}{3}}, \frac{1}{2}\}$, $\theta_2 = \frac{1}{2}$. Then after $K = \mathcal{O}\left(\left(\sqrt{\frac{L}{\mu}}\sqrt{\beta\left(\frac{\omega}{n} + 1\right)} + \beta\right)\log\frac{1}{\varepsilon}\right)$ iterations $\mathbb{E}\left[\Psi^K\right] \leq \varepsilon\Psi^0$, where, the Lyapunov function Ψ^K is defined as $\Psi^K := \mathbb{Z}^K + \mathbb{Y}^K + \mathbb{W}^K$.*
Total information, sent by `DHPL-Katyusha` with such parameters, is $\mathcal{O}\left(\left(\sqrt{\frac{L}{\mu}}\sqrt{\frac{1}{\beta}\left(\frac{\omega}{n} + 1\right)} + 1\right)\log\frac{1}{\varepsilon}\right)$.

2.3. RandK and PermK comparison

For RandK compressor, the exact ω is equal to $\frac{d}{K}$ (Beznosikov et al., 2024). Therefore, using Lemma 2.4 we get the efficient Lipschitz constant.

Lemma 2.8 (The efficient Lipschitz for RandK compressor in the horizontal case). *For `DHPL-Katyusha` with RandK compressor, the effective Lipschitz constant is less than:*

$$\widetilde{L}_{Rand} = \left(\frac{d}{nK} + 1\right)L$$

As Q_i for PermK operator are correlated, the analysis for it is slightly different from the original `DHPL-Katyusha`. But we still managed to find its efficient Lipschitz constant and therefore can estimate asymptotics for it.

Lemma 2.9 (The efficient Lipschitz for PermK compressor in the horizontal case). *For `DHPL-Katyusha` with PermK compressor, the effective Lipschitz constant is equal to L.*

As it can be seen in the above lemmas, $\widetilde{L}_{Rand}$ is generally greater than $\widetilde{L}_{Perm}$. Therefore, PermK compressor is more suitable, in terms of the number of iterations, for `DHPL-Katyusha` than RandK.

3. Vertical Case

In this section, we introduce the `Katyusha`-based algorithm of distributed optimization with vertical data division, which is named as `DVPL-Katyusha`. We introduce an algorithm for the problem of minimization of the MSE function. This can be mathematically written as:

$$\min_{x\in\mathbb{R}^d}\left[f(x) := \frac{1}{s}\|Ax - b\|^2\right], \quad (2)$$

where $b \in \mathbb{R}^s$ is a vector of targets, $A \in \mathbb{R}^{s \times d}$ is a feature matrix, s is a number of samples and d is a number of features.

In the vertical setup, we assume that every worker has its own set of features, which is formally represented, as a set of columns of the feature matrix A. Given the above, we can rewrite 2 in the form of:

$$\min_{x\in\mathbb{R}^d}\left[f(x) := \frac{1}{s}\left\|\sum_{i=1}^n A_i x_i - b\right\|^2\right], \quad (3)$$

where as A_i and x_i we denote submatrix and point subvector corresponding to the set of i-th worker's features, n is a number of devices.

The practical utility of the MSE loss function is significant, as linear models are quick to train and can serve as effective

benchmarks. Additionally, large models such as neural networks can leverage a trained model to generate features and then utilize only the final linear layer. Furthermore, implementing privacy mechanisms in vertical federated learning is simpler with this approach through linear models, like in (Huang et al., 2022).

Algorithm 2 `DVPL-Katyusha`

Input: initial $y^0 = \omega^0 = z^0 \in \mathbb{R}^d$, batch size b_s, step size $\eta = \frac{\theta_2}{(1+\theta_2)\theta_1}$, $\sigma = \frac{\mu}{L}$, where $\widetilde{L} = \frac{L}{b_s}\left(1 + (w-1)\frac{s \cdot \sum\limits_{j=1}^{s} L_j^2}{\mu^2}\right)$, parameters $\theta_1, \theta_2 \in \mathbb{R}$ and probability $p \in (0,1]$, every worker has the same random seed.

1: **for** $k = 0, 1, 2, \ldots K$ **do**
2:     **for** $i = 1 \ldots n$ in parallel **do**
3:         Select uniformly batch J^k, $|J^k| = b_s$
4:         $x_i^k \leftarrow \theta_1 z_i^k + \theta_2 \omega_i^k + (1 - \theta_1 - \theta_2) y_i^k$
5:         **for** $j \in J^k$ **do**
6:             Compute $\langle A_{ji}^T, x_i^k - w_i^k \rangle$
7:             Using communications
                    broadcast $Q_i\left(\langle A_{ji}^T, x_i^k - w_i^k \rangle\right)$
8:         **end for**
9:         $g_i^k \leftarrow \frac{2}{b_s} \sum\limits_{j \in J^k} A_{ji}^T \sum\limits_{i=1}^{n} Q_i\left(\langle A_{ji}^T, x_i^k - w_i^k \rangle\right)$
                    $+ \frac{2}{s}\left(A^T A w^k - A^T b\right)_i$
10:         $z_i^{k+1} \leftarrow \frac{1}{1+\eta\sigma}(\eta\sigma x_i^k + z_i^k - \frac{\eta}{L} g_i^k)$
11:         $y_i^{k+1} \leftarrow x_i^k + \theta_1(z_i^{k+1} - z_i^k)$
12:         $\omega_i^{k+1} \leftarrow \begin{cases} y_i^k, & \text{with probability } p \\ \omega_i^k, & \text{with probability } 1 - p \end{cases}$
13:         **if** $w_i^{k+1} = y_i^k$ **then**
14:             **for** $j = 1 \ldots s$ **do**
15:                 Compute $\langle A_{ji}^T, x_i^k - w_i^k \rangle$
16:                 Using communications
                        broadcast $\langle A_{ji}^T, x_i^k - w_i^k \rangle$
17:             **end for**
18:             Compute $\frac{2}{s}\left(A^T A w^k - A^T b\right)_i$
19:         **end if**
20:     **end for**
21: **end for**

3.1. Introduction of **DVPL-Katyusha**

In `DVPL-Katyusha` (Algorithm 2) every worker has its own set of features. Therefore, unlike the horizontal regime, in the vertical case all operations are performed on subvectors, corresponding to individual worker's components. We denote such subvectors with the same let-

ter as the vector itself, but with lower index i. Similar to `DHPL-Katyusha` (Algorithm 1) at each iteration every worker sets x_i^k as a convex combination of z_i^k, w_i^k and y_i^k (line 4). A gradient calculated on the j-th sample looks like $2(A_j^T \langle A_j^T, x^k \rangle - A_j^T b)$. This is why, to perform a step, we need to obtain a dot product $\langle A_j^T, x^k - w^k \rangle$ for every sample in a batch of a size b_s. We use batching in `DVPL-Katyusha` in order to reduce the number of algorithm's iterations, and to make it more efficient we choose samples uniformly. As every worker can compute only its own part of this sum (in our denotations for a single sample j it is $\langle A_{ji}^T, x_i^k - w_i^k \rangle$), all such parts from every batch sample should be broadcasted in order to compute g^k (line 6, line 7). To reduce communication complexity, we broadcast compressed scalars (line 7). After this, a standard momentum part of `Katyusha` is performed locally (line 10, line 11), after which w^k is updated with some probability, with every worker having the same random seed for it (line 12). If w^k is adjusted to the new point, the full gradient (e.g., all samples are used) in this vector is calculated the same way as is written above, but without any compression (line 18). Note that similar to the horizontal case, we consider probability of such update low, e.g., full gradient is calculated infrequently.

3.2. Convergence results

As in `DHPL-Katyusha` we need to assume our target function to be strong convex and worker's functions to be L-smooth. In terms of the MSE problem, this takes a form of:

Assumption 3.1 *The matrix $A^T A$ has all eigenvalues bounded in a segment of $[\mu, L]$.*

To estimate the asymptotics of the algorithm, we need to get a bound of the efficient Lipschitz constant for `DVPL-Katyusha`.

Lemma 3.2 *The efficient Lipschitz constant for* `DVPL-Katyusha` *is less than $\widetilde{L}$, where:*

$$\widetilde{L} = \frac{L}{b_s}\left(1 + (w-1)\frac{s \cdot \sum\limits_{j=1}^{s} L_j^2}{\mu^2}\right).$$

We also proved that under made assumptions there exists a function that has $\widetilde{L}$ proportional to $\frac{L^3}{\mu^2}$, therefore our asymptotics for `DVPL-Katyusha` is optimal in the terms of L and μ constants.

Lemma 3.3 *There exists such function that holds under 3.1 that its efficient Lipschitz constant $\widetilde{L}$ is proportional to $(\omega - 1)\frac{L^3}{\mu^2}$.*

Knowing the effective Lipschitz constant, we can estimate a convergence rate of our algorithm:

Theorem 3.4 *Let Assumption 3.1 be hold. Denote $\widetilde{L}$ as $\frac{L}{b_s}\left(1 + (w-1)\frac{s \cdot \sum_{j=1}^{s} L_j^2}{\mu^2}\right)$ and x^* as the solution for the problem 2. Then after k iterations of* `DVPL-Katyusha`

$$
\mathbb{E}\left[\mathbb{Z}^{k+1} + \mathbb{Y}^{k+1} + \mathbb{W}^{k+1}\right]
$$
$$
\leq \frac{1}{1+\eta\sigma}\mathbb{Z}^k + (1 - \theta_1(1-\theta_2))\mathbb{Y}^k + \left(1 - \frac{p\theta_1}{1+\theta_1}\right)\mathbb{W}^k,
$$

where:

$$
\mathbb{Z}^k := \frac{\widetilde{L}(1+\eta\sigma)}{2\eta}\|z^k - x^*\|^2,
$$
$$
\mathbb{Y}^k := \frac{1}{\theta_1}\left(f(y^k) - f(x^*)\right),
$$
$$
\mathbb{W}^k := \frac{\theta_2(1+\theta_1)}{p\theta_1}\left(f(\omega^k) - f(x^*)\right).
$$

After choosing concrete parameters, we can get an estimation for a total number of iterations and sent information.

Corollary 3.5 *Denote β as a density coefficient of compressor operator Q. Denote $L^2 = \frac{1}{s}\sum_{j=1}^{s} L_j^2$. Let $p = \frac{b_s}{s\beta}$, $\theta_1 = \min\{\sqrt{\frac{2\sigma\frac{s\beta}{b_s}}{3}}, \frac{1}{2}\}$, $\theta_2 = \frac{1}{2}$. Then, after $K = \mathcal{O}\left(\frac{s}{b_s}\left(\sqrt{\frac{L}{\mu}}\sqrt{\beta\left(\frac{1}{s} + (w-1)\frac{sL^2}{\mu^2}\right)} + \beta\right)\log\frac{1}{\varepsilon}\right)$ iterations $\mathbb{E}\left[\Psi^K\right] \leq \varepsilon\Psi^0$, where the Lyapunov function Ψ^K is defined as $\Psi^K := \mathbb{Z}^K + \mathbb{Y}^K + \mathbb{W}^K$.*
Total information, sent by `DVPL-Katyusha` *with such parameters, is $\mathcal{O}\left(\frac{s}{b_s}\left(\sqrt{\frac{L}{\mu}}\sqrt{\frac{1}{\beta}\left(\frac{1}{s} + (w-1)\frac{sL^2}{\mu^2}\right)} + 1\right)\log\frac{1}{\varepsilon}\right)$.*

The average information cost of a single iteration is $O\left(\frac{b_s}{\omega} + ps\right)$. Therefore, we choose p as $\frac{b_s}{s\omega}$ to asymptotically reduce it.

3.3. RandK and PermK comparison

In vertical `Katyusha`, beside basic scalar compression, we can also use a RandK compressor (Beznosikov et al., 2024), which compresses data by choosing random coordinates and sends only them. In our case, every worker will choose k samples from the batch of size b and do AllReduce with this data. Thus, we formulate a theorem:

Theorem 3.6 *The effective Lipschitz constant for RandK*

compressor is less than:

$$
\widetilde{L}_{Rand} = \frac{2L}{b_s}\left(\left(\frac{b_s}{K} - 1\right)\frac{s\sum_{j=1}^{s} L_j^2}{\mu^2} + 1\right).
$$

Beside RandK, we also implemented another compression strategy (Szlendak et al., 2021). In PermK every worker choose its own set of samples, which can not intersect with any other set. After that, each worker do AllReduce with calculated values. Therefore, we can prove a theorem:

Theorem 3.7 *The effective Lipschitz constant for PermK compressor is less than:*

$$
\widetilde{L}_{Perm} = 2L\frac{s\sum_{j=1}^{s} L_j^2}{\mu^2}.
$$

Unlike horizontal regime, PermK compressor does not have advantage in iteration complexity at RandK.

4. Numerical Experiments

In this section, we conduct multiple numerical experiments on horizontal and vertical cases with a binary classification problem and a strongly convex target function on `mushrooms`, `a5a` and `w3a` datasets from the LIBSVM library.

4.1. Binary classification in the horizontal regime

As assumed in our paper, in horizontal case we present our target function as the following finite sum:

$$
\min_{x \in \mathbb{R}^d}\left\{f(x) = \frac{1}{n}\sum_{m=1}^{n} f_m(x)\right\},
$$

where $f_m(x) = \frac{1}{s/n}\sum_{i=1}^{s/n}\log(1+\exp(-y_i a_i^T x)) + \frac{\lambda}{2}\|x\|_2^2$ – the sum here is taken with samples (a_i, y_i) from the local dataset of the m-th worker, $\lambda = L/100 - L$ here is the target function's $f(x)$ smoothness constant.

In our experiments on horizontal case, we have $n = 100$ devices involved in the training process and compare `DHPL-Katyusha` in combination with Rand1% and PermK compressors with accelerated algorithm `ADIANA` and vanilla GD and AGD, running in the same distributed setting. All of the plots on the Figure 1 are constructed for the convergence with tuned parameters. It is known from the article (Ernie Chan1 and van de Geijn1, 2006) that the AllReduce of vectors of size d takes time proportional to the magnitude of $2\frac{n-1}{n}d$, where d is the size of the vectors transmitted in this way. On the x-axis, we postponed either
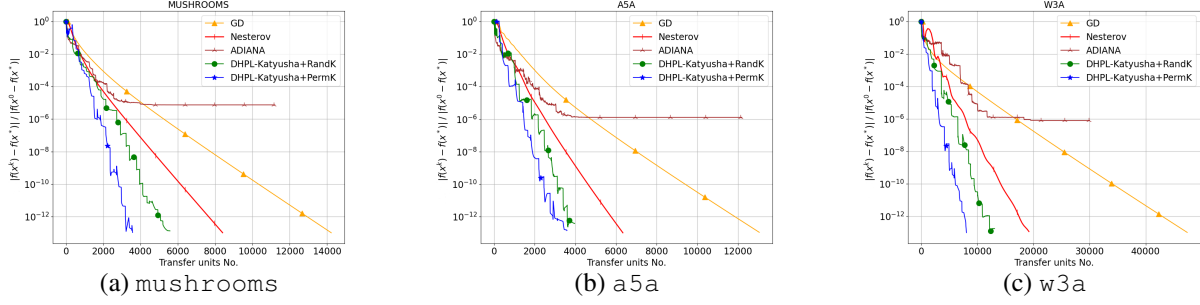
(a) `mushrooms`   (b) `a5a`   (c) `w3a`

Figure 1: Comparison of different algorithms for solving optimization problem from section 4.1 in horizontal case on LIBSVM datasets `mushrooms`, `a5a` and `w3a`.
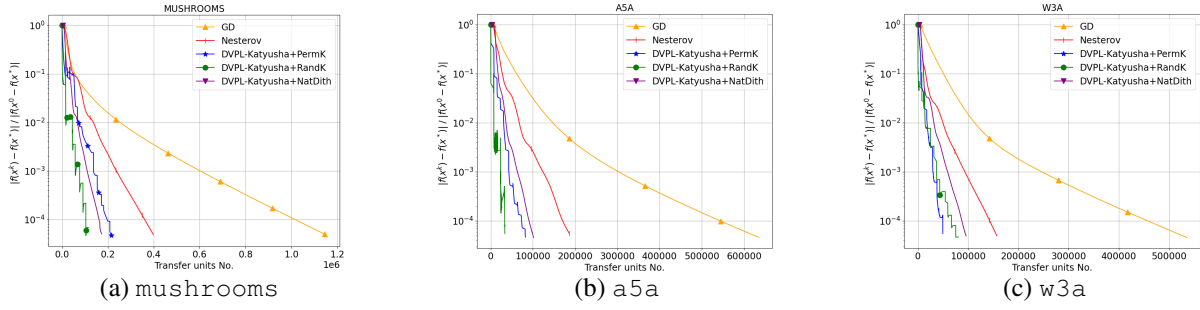

(a) `mushrooms`   (b) `a5a`   (c) `w3a`

Figure 2: Comparison of different algorithms for solving optimization problem from section 4.2 in vertical case on LIBSVM datasets `mushrooms`, `a5a` and `w3a`.

$\frac{d}{\beta}$ or d, depending on the outcome of the coinflip produced during the iteration. Thus, on the x-axis, we have a number proportional to the number of numpy.doubles (without taking into account the $2\frac{n-1}{n}$ multiplier that is the same for all the plots) transmitted during the entire learning process.

As it was expected in theory, our algorithm has an advantage over AGD with a total communication complexity of $\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\right)$ due to the n term in the denominator of our algorithm's best complexity – $\mathcal{O}\left(\sqrt{\frac{L}{n\mu}}\right)$. More than that, PermK clearly performs better than Rand1% in such setup, as can be seen from the graph.

Finally, it worth to be noted that the ADIANA algorithm in our setup could not be run to convergence, the algorithm converges only to the neighborhood of true solution, which is not stated in theory. At the same time, it is worth noting that a wide range of algorithm parameters were considered and some of its modifications were considered too in order to run it to the full convergence.

4.2. Linear regression in the vertical regime

In the case of vertical data partition, again, we present our target function as:

$$\min_{x\in\mathbb{R}^d}\left\{f(x)=\frac{1}{s}\sum_{i=1}^{s}l_i(a_i^T x,y_i)+\frac{\lambda}{2}\|x\|_2^2\right\},$$

where $a_i\in\mathbb{R}^d, y_i\in\{-1,1\}, \forall i\in 1,\ldots,s, \lambda=L/100$. $l_i:\mathbb{R}^2\to\mathbb{R}: l_i(a_i^T x,y_i)=(a_i^T x-y_i)^2$.

We also conducted experiments to compare `DVPL-Katyusha` in combination with Rand1%, Natural Dithering (Horváth et al., 2019) and PermK with vertical versions of regular gradient descent and its Nesterov acceleration in the same distributed setting. Number of workers in the experiments we conducted is $n=5$. And again, on the x-axis, we postpone either $\frac{b}{\beta}$ or s for each iteration, without taking into account the coefficient $2\frac{n-1}{n}$, which is the same for all implementations.

Overall, it is clear from the graph that there is a significant acceleration of convergence compared to the naive implementation of the Nesterov approach in a vertically distributed mode (when devices share all s values of their local models, after which they calculate their part of the gradient and produce a Nesterov descent based on it) and with the same implementation of ordinary gradient descent.

In addition, the superiority of the PermK compressor is no longer present in this case, and the Rand1% compressor comes out on top, which emphasizes the different importance of compressors' properties for different data division modes.

8

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Alacaoglu, A. and Malitsky, Y. (2022). Stochastic variance reduction for variational inequality methods.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. (2017). Qsgd: Communication-efficient sgd via gradient quantization and encoding.

Allen-Zhu, Z. (2016). Katyusha: The first direct acceleration of stochastic gradient methods.

Beznosikov, A. and Gasnikov, A. (2023). Similarity, compression and local steps: Three pillars of efficient communications for distributed variational inequalities.

Beznosikov, A., Horváth, S., Richtárik, P., and Safaryan, M. (2024). On biased compression for distributed learning.

Beznosikov, A., Richtárik, P., Diskin, M., Ryabinin, M., and Gasnikov, A. (2023). Distributed methods with compressed communication for solving variational inequalities, with theoretical guarantees.

Cai, D., Fan, T., Kang, Y., Fan, L., Xu, M., Wang, S., and Yang, Q. (2022). Accelerating vertical federated learning. *IEEE Transactions on Big Data*, page 1–10.

Castiglia, T., Das, A., Wang, S., and Patterson, S. (2023). Compressed-vfl: Communication-efficient learning with vertically partitioned data.

Chen, W., Ma, G., Fan, T., Kang, Y., Xu, Q., and Yang, Q. (2021). Secureboost+ : A high performance gradient boosting tree framework for large scale vertical federated learning.

Chilimbi, T., Suzue, Y., Apacible, J., and Kalyanaraman, K. (2014). Project adam: Building an efficient and scalable deep learning training system. In *11th USENIX symposium on operating systems design and implementation (OSDI 14)*, pages 571–582.

Ernie Chan1, , . H. A. P. and van de Geijn1, R. (2006). Collective communication: theory, practice, and experience. *Wiley InterScience*.

Gorbunov, E. (2021). Distributed and stochastic optimization methods with gradient compression and local steps.

Gorbunov, E., Burlachenko, K., Li, Z., and Richtárik, P. (2022). Marina: Faster non-convex distributed learning with compression.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. (2018). Accurate, large minibatch sgd: Training imagenet in 1 hour.

Gu, B., Xu, A., Huo, Z., Deng, C., and Huang, H. (2020). Privacy-preserving asynchronous federated learning algorithms for multi-party vertically collaborative learning.

Horváth, S., Kovalev, D., Mishchenko, K., Stich, S., and Richtárik, P. (2019). Stochastic distributed learning with gradient quantization and variance reduction.

Huang, L., Li, Z., Sun, J., and Zhao, H. (2022). Coresets for vertical federated learning: Regularized linear regression and k-means clustering.

Johnson, R. and Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. (2017). Federated learning: Strategies for improving communication efficiency.

Kovalev, D., Horvath, S., and Richtarik, P. (2019). Don't jump through hoops and remove those loops: Svrg and katyusha are better without the outer loop.

Li, Z., Bao, H., Zhang, X., and Richtárik, P. (2021). Page: A simple and optimal probabilistic gradient estimator for nonconvex optimization.

Li, Z., Kovalev, D., Qian, X., and Richtárik, P. (2020). Acceleration for compressed gradient descent in distributed and federated optimization.

Mishchenko, K., Gorbunov, E., Takáč, M., and Richtárik, P. (2023). Distributed learning with compressed gradient differences.

Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takáč, M. (2017). Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International conference on machine learning*, pages 2613–2621. PMLR.

Richtárik, P. and Takáč, M. (2013). Distributed coordinate descent method for learning with big data.

Smith, V., Forte, S., Ma, C., Takáč, M., Jordan, M. I., and Jaggi, M. (2018). Cocoa: A general framework for communication-efficient distributed optimization. *Journal of Machine Learning Research*, 18(230):1–49.

Sun, J., Xu, Z., Yang, D., Nath, V., Li, W., Zhao, C., Xu, D., Chen, Y., and Roth, H. R. (2023). Communication-efficient vertical federated learning with limited overlapping samples.

Szlendak, R., Tyurin, A., and Richtárik, P. (2021). Permutation compressors for provably faster distributed non-convex optimization.

Verbraeken, J., Wolting, M., Katzy, J., Kloppenburg, J., Verbelen, T., and Rellermeyer, J. S. (2020). A survey on distributed machine learning. *Acm computing surveys (csur)*, 53(2):1–33.

Xu, W., Fan, H., Li, K., and Yang, K. (2021). Efficient batch homomorphic encryption for vertically federated xgboost.

Zhang, Q., Gu, B., Deng, C., Gu, S., Bo, L., Pei, J., and Huang, H. (2021). Asysqn: Faster vertical federated learning algorithms with better computation resource utilization. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery amp; Data Mining*, KDD '21. ACM.

Supplementary Material

Contents

A. Auxiliary Lemmas

Lemma A.1 (Lemma 12 from (Kovalev et al., 2019))*. For a random vector $x \in \mathbb{R}^d$ and any $y \in \mathbb{R}^d$, the variance of y can be represented in a form:*

$$\mathbb{E}\left[\|x - \mathbb{E}[x]\|^2\right] = \mathbb{E}\left[\|x - \|y\|^2\right] - \mathbb{E}\left[\|\mathbb{E}[x] - y\|^2\right] \tag{4}$$

The next lemma is a form of Young's inequality:

Lemma A.2 *. Let $\beta = \frac{\eta\theta_1}{L(1-\eta\theta_1)}$. Then $\forall a, b \in \mathbb{R}^d$ the following inequality holds:*

$$\langle a, b \rangle \geq -\frac{\|a\|^2}{2\beta} - \frac{\beta\|b\|^2}{2} \tag{5}$$

B. Horizontal partition

Although this might seem trivial, we further rely on gradient unbiasedness. Thus,

Lemma B.1 *. In Algorithm 1 $\mathbb{E}\left[g^k\right] = \nabla f(x^k)$.*

Proof: From Algorithm 1 we get:

$$g^k := \frac{1}{n}\sum_{i=1}^{n} Q_i\left(\nabla f_i(x^k) - \nabla f_i(\omega^k)\right) + \nabla f(\omega^k) \tag{6}$$

Using Tower Property and Definition 1.1 of compressors' unbiasedness:

$$\mathbb{E}\left[g^k\right] := \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\mathcal{Q}_i\left(\nabla f_i(x^k) - \nabla f_i(\omega^k)\right)\right] + \nabla f(\omega^k) = \nabla f(x^k)$$

Lemma B.2 (Revised Lemma 6 from (Kovalev et al., 2019))*. In Algorithm 1 the following holds:*

$$\mathbb{E}\left[\|g^k - \nabla f(x^k)\|^2\right] \leq \frac{2L\omega}{n}\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right) \tag{7}$$

$$\tilde{L} = \frac{L\omega}{n}$$

Proof: From Algorithm 1, we derive 6, thus, with an empty term, after opening the square and taking advantage of the unbiased and independent compression operators in the scalar product:

$$
\begin{aligned}
\mathbb{E}\left[\|g^k - \nabla f(x^k)\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^{n} Q_i^w\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} + \underline{\nabla f(\omega^k)} - \underline{\nabla f(x^k)}\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^{n}\left(\mathcal{Q}_i\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} - \left\{\underline{\nabla f_i(x^k)} - \underline{\nabla f_i(\omega^k)}\right\}\right)\right\|^2\right] \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}\left[\left\|\mathcal{Q}_i\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} - \left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\}\right\|^2\right] \\
&\quad + \frac{1}{n^2}\sum_{i\neq l}\mathbb{E}[\langle\mathcal{Q}_i\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} - \left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\}, \mathcal{Q}_l\{\nabla f_l(x^k) \\
&\quad - \nabla f_l(\omega^k)\} - \left\{\nabla f_l(x^k) - \nabla f_l(\omega^k)\right\}\rangle]
\end{aligned}
$$

12

$$= \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{E} \left[\left\| \mathcal{Q}_i \left\{ \nabla f_i(x^k) - \nabla f_i(\omega^k) \right\} - \left\{ \nabla f_i(x^k) - \nabla f_i(\omega^k) \right\} \right\|^2 \right]$$

Next, using Assumption 2.1:

$$
\begin{aligned}
\mathbb{E} \left[\left\| g^k - \nabla f(x^k) \right\|^2 \right] &= \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{E} \left[\left\| \mathcal{Q}_i \left\{ \nabla f_i(x^k) - \nabla f_i(\omega^k) \right\} - \left\{ \nabla f_i(x^k) - \nabla f_i(\omega^k) \right\} \right\|^2 \right] \\
&\leq \frac{\omega}{n^2} \sum_{i=1}^{n} \mathbb{E} \left[\left\| \nabla f_i(x^k) - \nabla f_i(\omega^k) \right\|^2 \right]
\end{aligned}
$$

And finally, using Lipschitz property (Assumption 2.1) and convexity (Assumption 2.2) of functions $f_i(x)$:

$$
\begin{aligned}
\mathbb{E} \left[\left\| g^k - \nabla f(x^k) \right\|^2 \right] &\leq \frac{\omega}{n^2} \sum_{i=1}^{n} 2L \left(f_i(\omega^k) - f_i(x^k) - \langle \nabla f_i(x^k); \omega^k - x^k \rangle \right) \\
&\leq \frac{2L\omega}{n} \left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k); \omega^k - x^k \rangle \right)
\end{aligned}
$$

$\square$

Lemmas 7, 8, as already stated in the main section, are obtained from the original lemmas by replacing (only when $\frac{\omega}{n} > 1$) the Lipschitz constant of the gradient with a constant $\frac{\omega}{n}$ times greater than the one given by assumption 2.1. We now need to show that nothing changes in how lemmas 7, 8 behave at their core, although an Algorithm has experienced slight changes:

Lemma B.3 (Revised Lemma 7 from (Kovalev et al., 2019)). *In Algorithm 1 the following holds:*

$$\langle g^k, x^* - z^{k+1} \rangle + \frac{\mu}{2} \left\| x^k - x^* \right\|^2 \geq \frac{\widetilde{L}}{2\mu} \left\| z^k - z^{k+1} \right\|^2 + Z^{k+1} - \frac{1}{1+\eta\sigma} Z^k \tag{8}$$

Proof: We will only proof the lemma for the case $\frac{\omega}{n} > 1$, as the opposite case can be observed in the original paper. From Algorithm 1:

$$z^{k+1} := \frac{1}{1 + \eta\frac{\mu}{\widetilde{L}}} \left(\eta \frac{\mu}{\widetilde{L}} x^k + z^k - \frac{\eta}{\widetilde{L}} g^k \right)$$

$$\rightarrow g^k = \frac{\widetilde{L}}{\mu} \left(z^k - z^{k+1} \right) + \mu \left(x^k - z^{k+1} \right)$$

Thus (we will simply neglect the underlined summand):

$$
\begin{aligned}
\langle g^k, z^{k+1} - x^* \rangle &= \mu \langle x^k - z^{k+1}, z^{k+1} - x^* \rangle + \frac{\widetilde{L}}{\eta} \langle z^k - z^{k+1}, z^{k+1} - x^* \rangle \\
&= \frac{\mu}{2} \left(\| x^k - x^* \|^2 - \underline{\| x^k - z^{k+1} \|^2} - \| z^{k+1} - x^* \|^2 \right) \\
&\quad + \frac{\widetilde{L}}{2\eta} \left(\| z^k - x^* \|^2 - \| z^k - z^{k+1} \|^2 - \| z^{k+1} - x^* \|^2 \right) \\
&\leq \frac{\mu}{2} \| x^k - x^* \|^2 + \frac{\widetilde{L}}{2\eta} \left(\| z^k - x^* \|^2 - (1+\eta\sigma) \| z^{k+1} - x^* \|^2 \right) - \frac{\widetilde{L}}{2\eta} \| z^k - z^{k+1} \|^2
\end{aligned}
$$

Which concludes the proof $\square$

Now let us show that Lemma 8 can be rewritten in the same manner, replacing L - Lipschitz constant with the new $\widetilde{L}$.

Lemma B.4 (Revised Lemma 8 from (Kovalev et al., 2019)). *We have:*

$$\frac{1}{\theta_1} \left(f(y^{k+1}) - f(x^k) \right) - \frac{\theta_2}{2\widetilde{L}\theta_1} \left\| g^k - \nabla f(x^k) \right\|^2 \leq \frac{\widetilde{L}}{2\eta} \left\| z^{k+1} - z^k \right\|^2 + \langle g^k, z^{k+1} - z^k \rangle \tag{9}$$

13

Proof: In this lemma, again, we will only consider the case of $\frac{\omega}{n} > 1$. From Algorithm 1, one can see that $y^{k+1} \leftarrow x^k + \theta_1(z^{k+1} - z^k)$. Below we will introduce an empty term, use $\widetilde{L}$-smoothness of function $f(x)$ and, after that, utilize Lemma 5, where $\beta = \frac{\eta \theta_1}{\widetilde{L}(1 - \eta \theta_1)}$:

$$
\frac{\widetilde{L}}{2\eta} \|z^{k+1} - z^k\|^2 + \langle g^k, z^{k+1} - z^k \rangle = \frac{1}{\theta_1} \left(\frac{\widetilde{L}}{2\eta \theta_1} \|\theta_1 \left(z^{k+1} - z^k\right)\|^2 + \langle g^k, \theta_1 \left(z^{k+1} - z^k\right) \rangle \right)
$$

$$
= \frac{1}{\theta_1} \left(\frac{\widetilde{L}}{2\eta \theta_1} \|y^{k+1} - x^k\|^2 + \langle g^k, y^{k+1} - x^k \rangle \right)
$$

$$
= \underline{\frac{\widetilde{L}}{2\theta_1} \|y^{k+1} - x^k\|^2 + \frac{1}{\theta_1} \langle \nabla f(x^k), y^{k+1} - x^k \rangle}
$$

$$
+ \frac{\widetilde{L}}{2\theta_1} \left(\frac{1}{\eta \theta_1} \underline{-1} \right) \|y^{k+1} - x^k\|^2 + \frac{1}{\theta_1} \langle g^k \underline{-\nabla f(x^k)}, y^{k+1} - x^k \rangle)
$$

$$
= \frac{1}{\theta_1} \frac{\widetilde{L}}{2\theta_1} \|y^{k+1} - x^k\|^2 + \frac{1}{\theta_1} \langle \nabla f(x^k), y^{k+1} - x^k \rangle
$$

$$
+ \frac{\widetilde{L}}{2\theta_1} \left(\frac{1}{\eta \theta_1} - 1 \right) \|y^{k+1} - x^k\|^2 + \frac{1}{\theta_1} \langle g^k - \nabla f(x^k), y^{k+1} - x^k \rangle
$$

$$
\geq \frac{1}{\theta_1} \left(f(y^{k+1}) - f(x^k) \right) + \frac{\widetilde{L}}{2\theta_1} \left(\frac{1}{\eta \theta_1} - 1 \right) \|y^{k+1} - x^k\|^2
$$

$$
+ \frac{1}{\theta_1} \langle g^k - \nabla f(x^k), y^{k+1} - x^k \rangle
$$

$$
\geq \frac{1}{\theta_1} \left(f(y^{k+1}) - f(x^k) - \frac{\eta \theta_1}{2\widetilde{L}(1 - \eta \theta_1)} \|g^k - \nabla f(x^k)\|^2 \right)
$$

$$
= \frac{1}{\theta_1} \left(f(y^{k+1}) - f(x^k) - \frac{\theta_2}{2\widetilde{L}} \|g^k - \nabla f(x^k)\|^2 \right)
$$

The last equality was received due to $\eta = \frac{\theta_2}{(1 + \theta_2) \theta_1}$. Thus, we derive:

$$
\frac{1}{\theta_1} \left(f(y^{k+1}) - f(x^k) \right) - \frac{\theta_2}{2\widetilde{L}\theta_1} \|g^k - \nabla f(x^k)\|^2 \leq \frac{\widetilde{L}}{2\eta} \|z^{k+1} - z^k\|^2 + \langle g^k, z^{k+1} - z^k \rangle
$$

$\square$

Lemma 9, however, does not require any changes in the proof at all, because it is based only on the definition of ω^{k+1} from Algorithm 1.

Lemma B.5 (Lemma 9 from (Kovalev et al., 2019)). *We have:*

$$
\mathbb{E}\left[f(\omega^{k+1}) \right] = (1 - p) f(\omega^k) + p f(y^k) \tag{10}
$$

Lemma B.6 (Revised Lemma 10 from (Kovalev et al., 2019)). *Considering lemmas B.2–B.5, we get:*

$$
\mathbb{Z}^k \left[\frac{1}{1 + \eta \sigma} \right] + \mathbb{Y}^k \left[(1 - \theta_1(1 - \theta_2)) \right] + \mathbb{W}^k \left[1 - \frac{p \theta_1}{1 + \theta_1} \right] \geq \mathbb{E}\left[\mathbb{Z}^{k+1} + \mathbb{Y}^{k+1} + \mathbb{W}^{k+1} \right] \tag{11}
$$

Proof: In this lemma, there is a significant difference between the $\frac{\omega}{n} < 1$ and $\frac{\omega}{n} > 1$ regimes. Further, we will perform the proof in both cases.

Case $\frac{\omega}{n} < 1$:

From Lemma B.1 we get $\mathbb{E}\left[g^k \right] = \nabla f(x^k)$, therefore, the following is inherited directly from the original proof:

$$
f(x^*) \geq f(x^k) + \frac{\theta_2}{\theta_1} \langle \nabla f(x^k), x^k - \omega^k \rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1} \left(f(x^k) - f(y^k) \right)
$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma} Z^k \right] + \mathbb{E}\left[\frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k) \right) - \frac{\theta_2}{2L\theta_1}\left\| g^k - \nabla f(x^k) \right\|^2 \right]$$

Using Lemma B.2, we derive:

$$
\begin{aligned}
f(x^*) \quad \geq \quad & f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k \rangle \\
+ \quad & \frac{1-\theta_1-\theta_2}{\theta_1}\left(f(x^k) - f(y^k) \right) + \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k \right] \\
+ \quad & \mathbb{E}\left[\frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k) \right) - \frac{\theta_2\omega}{\theta_1 n}\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k); \omega^k - x^k \rangle \right) \right] \\
= \quad & \frac{\theta_2}{\theta_1}\left(1 - \frac{\omega}{n} \right)\langle \nabla f(x^k), x^k - \omega^k \rangle - \frac{\theta_2}{\theta_1}\frac{\omega}{n}f(\omega^k) - \frac{\theta_2}{\theta_1}\left(1 - \frac{\omega}{n} \right)f(x^k) \\
- \quad & \frac{1-\theta_1-\theta_2}{\theta_1}f(y^k) + \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}f(y^{k+1}) \right]
\end{aligned}
$$

When ω is relatively small, we can use the convexity (Assumption 2.2) of function $f(x)$, and this will instantly end the proof (just as in the original paper):

$$
\begin{aligned}
f(x^*) \quad \geq \quad & \frac{\theta_2}{\theta_1}\left(1 - \frac{\omega}{n} \right)\left(f(x^k) - f(\omega^k) \right) - \frac{\theta_2}{\theta_1}\frac{\omega}{n}f(\omega^k) - \frac{\theta_2}{\theta_1}\left(1 - \frac{\omega}{n} \right)f(x^k) \\
- \quad & \frac{1-\theta_1-\theta_2}{\theta_1}f(y^k) + \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}f(y^{k+1}) \right] \\
= \quad & -\frac{\theta_2}{\theta_1}f(\omega^k) - \frac{1-\theta_1-\theta_2}{\theta_1}f(y^k) + \left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}f(y^{k+1}) \right]
\end{aligned}
$$

As one can observe, the same inequality as in the original paper has been obtained, thus deriving the same final expression:

$$\mathbb{Z}^k\left[\frac{1}{1+\eta\sigma} \right] + \mathbb{Y}^k\left[(1 - \theta_1(1-\theta_2)) \right] + \mathbb{W}^k\left[1 - \frac{p\theta_1}{1+\theta_1} \right] \geq \mathbb{E}\left[\mathbb{Z}^{k+1} + \mathbb{Y}^{k+1} + \mathbb{W}^{k+1} \right] \tag{12}$$

Case $\frac{\omega}{n} > 1$: In the case when the constant ω prevails, we are forced to repeat the proof of Lemma 10 from original paper to show that no transition is actually violated. Using strong convexity (Assumption 2.2) and adding an empty term, we get:

$$
\begin{aligned}
f(x^*) \quad \geq \quad & f(x^k) + \langle \nabla f(x^k), x^* - x^k \rangle + \frac{\mu}{2}\|x^k - x^*\|^2 \\
\geq \quad & f(x^k) + \langle \nabla f(x^k), x^* \underline{-z^k + z^k} - x^k \rangle + \frac{\mu}{2}\|x^k - x^*\|^2
\end{aligned}
$$

From Algorithm 1, $z^k - x^k := \frac{\theta_2}{\theta_1}\left(x^k - \omega^k \right) + \frac{1-\theta_1-\theta_2}{\theta_1}\left(x^k - y^k \right)$, thus:

$$
\begin{aligned}
f(x^*) \quad \geq \quad & f(x^k) + \langle \nabla f(x^k), x^* \underline{-z^k + z^k} - x^k \rangle + \frac{\mu}{2}\|x^k - x^*\|^2 \\
\geq \quad & f(x^k) + \frac{\mu}{2}\|x^k - x^*\|^2 + \langle \nabla f(x^k), x^* - z^k \rangle \\
+ \quad & \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k \rangle + \underline{\frac{1-\theta_1-\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - y^k \rangle}
\end{aligned}
$$

Applying convexity to the underlined term, utilizing unbiasedness of the gradient and adding an empty term, we derive:

$$f(x^*) \quad \geq \quad f(x^k) + \frac{\mu}{2}\|x^k - x^*\|^2 + \langle \nabla f(x^k), x^* - z^k \rangle$$

15

$$+ \quad \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\langle \nabla f(x^k), x^k - y^k\rangle$$

$$\geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[\frac{\mu}{2}\|x^k - x^*\|^2\langle g^k, x^* \underline{-z^{k+1}}\rangle + \langle g^k, \underline{z^{k+1}} - z^k\rangle\right]$$

$$= \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[\frac{\mu}{2}\|x^k - x^*\|^2 + \langle g^k, x^* - z^{k+1}\rangle +^k, z^{k+1} - z^k\rangle\right]$$

Using $\frac{\omega}{n} > 1$ version of Lemma B.3, we get:

$$f(x^*) \quad = \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[\frac{\mu}{2}\|x^k - x^*\|^2 + \langle g^k, x^* - z^{k+1}\rangle + \langle g^k, z^{k+1} - z^k\rangle\right]$$

$$\geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k\right] + \mathbb{E}\left[\langle g^k, z^{k+1} - z^k\rangle + \frac{\widetilde{L}}{2\eta}\|z^k - z^{k+1}\|^2\right]$$

$$= \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k\right] + \mathbb{E}\left[\underline{\langle g^k, z^{k+1} - z^k\rangle + \frac{\widetilde{L}}{2\eta}\|z^k - z^{k+1}\|^2}\right]$$

Utilizing $\frac{\omega}{n} > 1$ version of Lemma B.4, we transform an underlined summand:

$$f(x^*) \quad = \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k\right] + \mathbb{E}\left[\underline{\langle g^k, z^{k+1} - z^k\rangle + \frac{\widetilde{L}}{2\eta}\|z^k - z^{k+1}\|^2}\right]$$

$$\geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k\right] + \mathbb{E}\left[\underline{\frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k)\right) - \frac{\theta_2}{2\widetilde{L}\theta_1}\|g^k - \nabla f(x^k)\|^2}\right]$$

$$\geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k + \frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k)\right)\right] - \mathbb{E}\left[\frac{\theta_2}{2\widetilde{L}\theta_1}\|g^k - \nabla f(x^k)\|^2\right]$$

Finally, using Lemma B.2 to evaluate the last summand, we get:

$$f(x^*) \quad \geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1 + \eta\sigma}Z^k + \frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k)\right)\right] - \mathbb{E}\left[\frac{\theta_2}{2\widetilde{L}\theta_1}\|g^k - \nabla f(x^k)\|^2\right]$$

$$\geq \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle \nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k)\right)\right]$$

$$- \quad \frac{\theta_2}{2\widetilde{L}\theta_1}\mathbb{E}\left[2\widetilde{L}\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right)\right]$$

$$= \quad f(x^k) + \frac{\theta_2}{\theta_1}\langle\nabla f(x^k), x^k - \omega^k\rangle + \frac{1 - \theta_1 - \theta_2}{\theta_1}\left(f(x^k) - f(y^k)\right)$$

$$+ \quad \mathbb{E}\left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}\left(f(y^{k+1}) - f(x^k)\right)\right]$$

$$- \quad \frac{\theta_2}{\theta_1}\mathbb{E}\left[f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right]$$

$$= \quad -\frac{\theta_2}{\theta_1}f(\omega^k) - \frac{1 - \theta_1 - \theta_2}{\theta_1}f(y^k) + \left[Z^{k+1} - \frac{1}{1+\eta\sigma}Z^k + \frac{1}{\theta_1}f(y^{k+1})\right]$$

As one can see, this result corresponds exactly to the formulation of Lemma 10 from the original article, with the only difference that now everywhere we have the constant $\widetilde{L}$ instead of L, which leads us to the same final expression:

$$\mathbb{Z}^k\left[\frac{1}{1+\eta\sigma}\right] + \mathbb{Y}^k\left[(1 - \theta_1(1 - \theta_2))\right] + \mathbb{W}^k\left[1 - \frac{p\theta_1}{1+\theta_1}\right] \geq \mathbb{E}\left[\mathbb{Z}^{k+1} + \mathbb{Y}^{k+1} + \mathbb{W}^{k+1}\right] \tag{13}$$

$\square$

From Lemma B.6 we get our main Theorem directly by substituting into the inequality the specified values that preserve the inequality:

Theorem B.7 (Theorem 11 from (Kovalev et al., 2019)). *Let Assumptions 2.1, 2.2 be hold. Additionally, let $p = \frac{1}{n}$, $\theta_1 = \min\{\sqrt{\frac{2\sigma n}{3}}, \frac{1}{2}\}$, $\theta_2 = \frac{1}{2}$. Then $\mathbb{E}\left[\Psi^k\right] \leq \epsilon\Psi^0$ after $K = \mathcal{O}\left(\left(n + \sqrt{\frac{\omega L}{\mu}}\right)\log\frac{1}{\epsilon}\right)$ iterations.*

Theorem B.8 (The efficient Lipschitz for PermK compressor in the horizontal case). *For* DHPL-Katyusha *with PermK compressor, the effective Lipschitz constant is equal to L.*

Proof: Denote Q_i as a PermK compressor that operates on i-th function. Therefore, we get:

$$\mathbb{E}\left[\left\|g^k - \nabla f(x^k)\right\|^2\right] = \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n Q_i^w\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} + \nabla f(\omega^k) - \nabla f(x^k)\right\|^2\right]$$

$$= \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n Q_i\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\} - \frac{1}{n}\sum_{i=1}^n\left\{\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\}\right\|^2\right]$$

$$\leq \mathbb{E}\left[A\frac{1}{n}\sum_{i=1}^n\left\|\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\|^2 - B\left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(x^k) - \nabla f_i(\omega^k)\right\|^2\right]$$

$$\leq \mathbb{E}\left[A\frac{1}{n}\sum_{i=1}^n\left\|\nabla f_i(x^k) - \nabla f_i(\omega^k)\right\|^2\right]$$

$$\leq 2LA\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k), \omega^k - x^k\rangle\right)$$

The first inequality is obtained by using A-B inequality from (Szlendak et al., 2021), the second inequality we get by utilizing the positive definiteness of the norm and the last inequality is obtained by using an analogy with B.1.

In the case $n \geq d$ we have $A = \frac{d-1}{n-1} \leq 1$, and in the case $n < d$ we have $A = 1$. Combining these cases, we get the estimation for the effective Lipschitz constant as 1.

C. Vertical case

In this section, we provide the complete proofs for our results in the vertical case.

Lemma C.1 (Revised Lemma 6 from (Kovalev et al., 2019))**.** *In Algorithm 2 the following holds:*

$$
\begin{aligned}
\|g^k(x^k) - \nabla f(x^k)\|^2 &\leq 2L\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k); \omega^k - x^k \rangle\right) \\
&\qquad \cdot \frac{1}{b}\left(1 + (w-1)\frac{s \cdot \sum_{j=1}^{s} L_j^2}{\mu^2}\right)
\end{aligned}
$$

Proof:

$$
\begin{aligned}
\mathbb{E}\left[\|g^k(x^k) - \nabla f(x^k)\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{b}\sum_{j\in\{J\}}\left[2\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right]A_j^T + \nabla f(\omega^k) - \nabla f(x^k)\right\|^2\right] \\
&\leq \frac{1}{b^2}\cdot b\sum_{j\in J}\cdot\mathbb{E}\left[\left\|2\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)A_j^T + \nabla f(\omega^k) - \nabla f(x^k)\right\|^2\right]
\end{aligned}
$$

Here we used a Cauchy-Bunyakovsky-Schwarz inequality. The next step is to evaluate the term within mathematic expectancy.

$$
\mathbb{E}\left[\left\|2\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)A_j^T + \nabla f(\omega^k) - \nabla f(x^k)\right\|^2\right] = \mathbb{E}\left[\left\|2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right) - \frac{2}{s}A^TA(x^k - w^k)\right\|^2\right]
$$

$$
\begin{aligned}
&= \mathbb{E}\left[\left\|2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right\|^2\right] + \mathbb{E}\left[\left\|\frac{2}{s}A^TA(x^k - w^k)\right\|^2\right] \\
&\quad - 2\cdot\mathbb{E}\left[\langle 2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right), \frac{2}{s}A^TA(x^k - w^k)\rangle\right] \\
&= \mathbb{E}\left[\left\|2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right\|^2\right] - \left\|\frac{2}{s}A^TA(x^k - w^k)\right\|^2 \leq \mathbb{E}\left[\left\|2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right\|^2\right]
\end{aligned}
$$

Now, using norm properties and Assumption 1.1 we get:

$$
\mathbb{E}\left[\left\|2A_j^T\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right\|^2\right] = 4\mathbb{E}\left[\|A_j^T\|^2\left|\sum_{i=1}^{n}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right|^2\right]
$$

$$
= 4\mathbb{E}\left[\|A_j^T\|^2\left|\sum_{i=1}^{n}\left(Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\right)^2 + \sum_{i\neq t}Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)Q_t\left(\langle a_j^t, x_t^k - \omega_t^k\rangle\right)\right|\right]
$$

$$
\leq 4\mathbb{E}\left[\|A_j^T\|^2\left|\sum_{i=1}^{n}\omega(\langle a_j^i, x_i^k - \omega_i^k\rangle)^2 + \sum_{i\neq t}\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)\left(\langle a_j^t, x_t^k - \omega_t^k\rangle\right)\right|\right]
$$

$$
= \quad 4\mathbb{E}\left[\left\|A_j^T\right\|^2 \left|\sum_{i=1}^n (\omega - 1)(\langle a_j^i, x_i^k - \omega_i^k\rangle)^2 + (\langle A_j^T, x^k - w^k\rangle)^2\right|\right]
$$

$$
= \quad \mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + 4(\omega - 1)\left\|A_j^T\right\|^2 \sum_{i=1}^n \left|\langle a_j^i, x_i^k - \omega_i^k\rangle\right|^2\right]
$$

$$
\leq \quad \mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + 4(\omega - 1)\left\|A_j^T\right\|^4 \left\|x^k - w^k\right\|^2\right]
$$

$$
= \quad \mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + 4(\omega - 1)\left\|A_j^T\right\|^4 \left\|\left(\frac{2}{s}A^TA\right)^{-1}(\nabla f(x^k) - \nabla f(w^k))\right\|^2\right]
$$

$$
\leq \quad \mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + (\omega - 1)\left\|A_j^T\right\|^4 s^2 \left\|\left(A^TA\right)^{-1}\right\|^2 \left\|\nabla f(x^k) - \nabla f(w^k)\right\|^2\right]
$$

$$
\leq \quad \mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + (\omega - 1)\left\|A_j^T\right\|^4 \frac{s^2}{\mu^2}\left\|\nabla f(x^k) - \nabla f(w^k)\right\|^2\right]
$$

$$
\leq \quad \left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k); \omega^k - x^k\rangle\right)\left(2L + 2L(w-1)\frac{s^2 \cdot \frac{\sum_{j=1}^s L_j^2}{s}}{\mu^2}\right)
$$

$$
= \quad 2L\left(f(\omega^k) - f(x^k) - \langle \nabla f(x^k); \omega^k - x^k\rangle\right)\left(1 + (w-1)\frac{s \cdot \sum_{j=1}^s L_j^2}{\mu^2}\right)
$$

Where the first equality is obtained by using absolute homogeneity of vector norm and having that $\sum_{i=1}^n Q_i\left(\langle a_j^i, x_i^k - \omega_i^k\rangle\right)$ is a scalar value, the first inequality is obtained by using compressor property 1.1, the second and the third inequalities are obtained by using the definition of the norm as the supremum, the fourth inequality is obtained by using that the spectral norm of matrix is bounded by the biggest eigenvalue, which is bounded by $\frac{1}{\mu}$, final inequality is obtained by using the Lipschitz property 2.1.

Lemma C.2 *There exists such function that holds under 3.1 that its efficient Lipschitz constant $\widetilde{L}$ is proportional to $(\omega - 1)\frac{L^3}{\mu^2}$.*

Proof:

Assume the size of batch $b = 1$. Therefore, in previous lemma we have an exact quality:

$$
\mathbb{E}\left[\|g^k - \nabla f(x^k)\|^2\right] = 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + (\omega - 1)\left\|A_j^T\right\|^2 \sum_{i=1}^n \left|\langle a_j^i, x_i^k - \omega_i^k\rangle\right|^2\right]
$$

Now assume matrix A is given as ($b < a$):

$$
A = \begin{pmatrix} a & a \\ b & -b \end{pmatrix}
$$

Assume that $k = 2$ and on the first iteration of `DVPL-Katyusha` ∇f_2 was chosen and b (from $Ax - b$) and x_0 are collinear with A_2^T.

Denote $x^k - w^k = (c, -c)^T$. It has its first coordinate equal to the negative second, because it is orthogonal to A_1^T.

As A_1^T is orthogonal with $x^k - w^k$:

$$
\nabla f_1(x^k) - \nabla f_1(w^k) = A_1^T \langle A_1^T, x^k - w^k\rangle = 0
$$

Therefore, we have the difference of the target gradient in x^k and in w^k to be equal to:

$$\nabla f(x^k) - \nabla f(w^k) = \nabla f_1(x^k) - \nabla f_1(w^k) + \nabla f_2(x^k) - \nabla f_2(w^k) = \nabla f_2(x^k) - \nabla f_2(w^k)$$

Note that the norm of $w^k - x^k$ equals to:

$$||w^k - x^k||^2 = 2c^2 = \frac{||\nabla f_2(x^k) - \nabla f_2(w^k)||^2}{4||A_2^T||^4} = \frac{||\nabla f(x^k) - \nabla f(w^k)||^2}{4||A_2^T||^4}$$

Also note that eigenvalues of $A^T A$ are equal to $2a^2$ and $2b^2$. Therefore $L = 2a^2$ and $\mu = 2b^2$.

Combining all of these facts we get:

$$
\begin{aligned}
\mathbb{E}\left[\|g^k - \nabla f(x^k)\|^2\right] &= 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2 + (\omega - 1)\left\|A_j^T\right\|^2 \sum_{i=1}^{n} |\langle a_j^i, x_i^k - \omega_i^k\rangle|^2\right] \\
&= 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2\right] + (\omega - 1)\left\|A_1^T\right\|^2 \left(a^2 c^2 + a^2 c^2\right) + (\omega - 1)\left\|A_2^T\right\|^2 \left(b^2 c^2 + b^2 c^2\right) \\
&= 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2\right] + 2(\omega - 1)\left\|A_1^T\right\|^4 c^2 + 2(\omega - 1)\left\|A_2^T\right\|^4 c^2 \\
&= 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2\right] + 2(\omega - 1)\left\|A_1^T\right\|^4 c^2 + 2(\omega - 1)\left\|A_1^T\right\|^4 c^2 \\
&= 4\mathbb{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(w^k)\right\|^2\right] + (\omega - 1)\frac{L^2}{2\mu^2}||\nabla f(x^k) - \nabla f(w^k)||^2 + \frac{1}{2}(\omega - 1)||\nabla f(x^k) - \nabla f(w^k)||^2
\end{aligned}
$$

Using the Lipschitz property in analogue with C.1, we conclude the proof.

C.1. RandK compressor

In this section we introduce a distributed algorithm with a RandK compressor. Every worker will independently choose it's own k samples from a batch with a size b and do standard AllReduce from algorithm only with them. Denote I_{ij} as an indicator of j-th worker choosing i-th component. Thus, the analogue of lemma 6 will look like:

$$
\begin{aligned}
\mathbb{E}\left\|g^k - \nabla f(x^k)\right\|^2 &= \mathbb{E}\left\|2\frac{1}{k}\sum_{i=1}^{b} A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{2}{s}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 4\mathbb{E}\left\|\frac{1}{k}\sum_{i=1}^{b} A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{s}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 4\mathbb{E}\left\|\sum_{i=1}^{b}\left[\frac{1}{k}A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{bs}A^T A\left(x^k - w^k\right)\right]\right\|^2 \\
&\leq 8\sum_{i=1}^{b}\mathbb{E}\left\|\frac{1}{k}A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{bs}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 8\sum_{i=1}^{b}\mathbb{E}\left[\left\|\frac{1}{k}A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij}\right\|^2 - \left\|\frac{1}{bs}A^T A\left(x^k - w^k\right)\right\|^2\right]
\end{aligned}
$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\left\|\frac{1}{k}A_i^T\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij}\right\|^2 - \left\|\frac{1}{bs}A^TA\left(x^k - w^k\right)\right\|^2\right]$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\left(\sum_{j=1}^{n}\langle a_j^i, x_i^k - \omega_i^k\rangle I_{ij}\right)^2\left\|\frac{1}{k}A_i^T\right\|^2 - \left\|\frac{1}{bs}A^TA\left(x^k - w^k\right)\right\|^2\right]$$

$$\leq \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\left(\sum_{j=1}^{n}(\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij})^2 + \sum_{j\neq t}(\langle a_j^i, x_i^k - \omega_i^k\rangle I_{ij})(\langle a_t^i, x_i^k - \omega_i^k\rangle I_{it})\right)\left\|\frac{1}{k}A_i^T\right\|^2\right] =$$

The first inequality was obtained by using CBS inequality.
Using the tower property with RandK random we get:

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\left(\sum_{j=1}^{n}\frac{k}{b}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2 + \sum_{j\neq t}\frac{k}{b}\frac{k}{b}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle\langle a_t^i, x_i^k - \omega_i^k\rangle\right)\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\frac{k}{b}\frac{k}{b}\left(\frac{b}{k}\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2 + \sum_{j\neq t}\langle a_j^i, a_j^i, x_{ji}^k - \omega_{ji}^k\rangle\langle a_t^i, x_i^k - \omega_i^k\rangle\right)\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\frac{k^2}{b^2}\left[\left(\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2 + \sum_{j\neq t}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle\langle a_t^i, x_i^k - \omega_i^k\rangle\right)\left\|\frac{1}{k}A_i^T\right\|^2 + \left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\frac{k^2}{b^2}\left[\left(\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle\right)^2\left\|\frac{1}{k}A_i^T\right\|^2 + \left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$= \quad 8\sum_{i=1}^{b}\mathbb{E}\left[\frac{1}{b^2}\left\|A_i^T\langle A_i^T, x_i^k - w_i^k\rangle\right\|^2\right] + 8\sum_{i=1}^{b}\mathbb{E}\left[\frac{k^2}{b^2}\left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$= \quad 2\sum_{i=1}^{b}\mathbb{E}\left[\frac{1}{b^2}\left\|\nabla f_i(x^k) - \nabla f_i(w^k)\right\|^2\right] + 8\sum_{i=1}^{b}\mathbb{E}\left[\frac{k^2}{b^2}\left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

$$\leq \quad \frac{4L}{b}\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right) + 8\sum_{i=1}^{b}\mathbb{E}\left[\frac{k^2}{b^2}\left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right]$$

Using the results obtained from Lemma C.1 we get an estimation for the second term:

$$8\sum_{i=1}^{b}\mathbb{E}\left[\frac{k^2}{b^2}\left(\frac{b}{k} - 1\right)\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2\left\|\frac{1}{k}A_i^T\right\|^2\right] \leq \quad 16L\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right)\left(\frac{1}{b}\left(\frac{b}{k} - 1\right)\frac{s\sum_{j=1}^{s}L_j^2}{4\mu^2}\right)$$

Thus, we get the final estimation:

$$\mathbb{E}\left\|g^k - \nabla f(x^k)\right\|^2 \leq \frac{4L}{b}\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right)\left(\left(\frac{b}{k} - 1\right)\frac{s\sum_{j=1}^{s}L_j^2}{\mu^2} + 1\right)$$

21

C.2. PermK compressor

In PermK compressor, each worker choose it's own sample and will AllReduce its components only from it.

$$
\begin{aligned}
\mathbb{E}\left\|g^k - \nabla f(x^k)\right\|^2 &= \mathbb{E}\left\|2\frac{n}{n}\sum_{i=1}^{n} A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{2}{s}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 4\mathbb{E}\left\|\sum_{i=1}^{n} A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{s}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 4\mathbb{E}\left\|\sum_{i=1}^{n}\left[A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{ns}A^T A\left(x^k - w^k\right)\right]\right\|^2 \\
&\leq 8\sum_{i=1}^{n}\mathbb{E}\left\|A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij} - \frac{1}{ns}A^T A\left(x^k - w^k\right)\right\|^2 \\
&= 8\sum_{i=1}^{n}\mathbb{E}\left[\left\|A_i^T \sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij}\right\|^2 - \left\|\frac{1}{ns}A^T A\left(x^k - w^k\right)\right\|^2\right] \\
&\leq 8\sum_{i=1}^{n}\mathbb{E}\left[\left(\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij}\right)^2 \left\|A_i^T\right\|^2\right] \\
&= 8\sum_{i=1}^{n}\mathbb{E}\left[\left(\sum_{j=1}^{n}(\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij})^2 + \sum_{j\neq t}(\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle I_{ij})(\langle a_t^i, x_{ti}^k - \omega_{ti}^k\rangle I_{it})\right) \left\|A_i^T\right\|^2\right] =
\end{aligned}
$$

Using that $\mathbb{E}I_{ij}^2 = \frac{1}{n}$ and as any sample can be chosen only by one worker, we get that $\mathbb{E}I_{it}I_{ij} = 0, j \neq t$. Therefore, using the analogy with C.1 we get:

$$
\begin{aligned}
&= 8\sum_{i=1}^{n}\mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n}\langle a_j^i, x_{ji}^k - \omega_{ji}^k\rangle^2 \left\|A_i^T\right\|^2\right] \\
&\leq 4L\left(f(\omega^k) - f(x^k) - \langle\nabla f(x^k); \omega^k - x^k\rangle\right)\frac{s\sum_{j=1}^{s}L_j^2}{\mu^2}
\end{aligned}
$$