Variance Reduced ProxSkip: Algorithm, Theory and Application to Federated Learning

Grigory Malinovsky
KAUST
grigory.malinovsky@kaust.edu.sa

Kai Yi
KAUST
kai.yi@kaust.edu.sa

Peter Richtárik
KAUST
peter.richtarik@kaust.edu.sa

Abstract

We study distributed optimization methods based on the *local training (LT)* paradigm: achieving communication efficiency by performing richer local gradient-based training on the clients before parameter averaging. Looking back at the progress of the field, we *identify 5 generations of LT methods*: 1) heuristic, 2) homogeneous, 3) sublinear, 4) linear, and 5) accelerated. The 5th generation, initiated by the ProxSkip method of Mishchenko et al. [2022] and its analysis, is characterized by the first theoretical confirmation that LT is a communication acceleration mechanism. Inspired by this recent progress, we contribute to the 5th generation of LT methods by showing that it is possible to enhance them further using *variance reduction*. While all previous theoretical results for LT methods ignore the cost of local work altogether, and are framed purely in terms of the number of communication rounds, we show that our methods can be substantially faster in terms of the *total training cost* than the state-of-the-art method ProxSkip in theory and practice in the regime when local computation is sufficiently expensive. We characterize this threshold theoretically, and confirm our theoretical predictions with empirical results.

1 Introduction

Announced in April 2017 in a Google AI blog [McMahan and Ramage, 2017], and citing four foundational papers [Konečný et al., 2016b,a, McMahan et al., 2017, Bonawitz et al., 2017] of what was to become a new and rapidly growing interdisciplinary field, *federated learning* (FL) constitutes a novel paradigm for training supervised machine learning models. The key idea is the acknowledgement that increasing amounts of data are being captured and stored on edge devices, such as mobile phones, sensors and hospital workstations, and that moving the data to a datacenter for centralized processing may be infeasible or undesirable for various reasons, including high energy costs and data privacy concerns [Kairouz et al., 2019, Li et al., 2020a]. FL faces a multitude of challenges which are being actively addressed by the research community.

1.1 Formalism

We study the standard optimization formulation of federated learning [Konečný et al., 2016a, McMahan et al., 2017, Kairouz et al., 2019, Wang et al., 2021] given by

$$\min_{x\in\mathbb{R}^{d'}} \phi(x), \qquad \phi(x) \coloneqq \sum_{i=1}^{M} \frac{n_i}{n}\phi_i(x), \qquad \phi_i(x) \coloneqq \frac{1}{n_i}\sum_{j=1}^{n_i}\phi_{ij}(x), \tag{1}$$

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

where M is the number of clients (devices, machines, workers), n_i is the number of training data points on client $i \in \{1, 2, \ldots, M\}$, and $n := \sum_{i=1}^M n_i$ is the total number of training data points collectively owned by this federation of M clients. Note that ϕ is the empirical risk over the federated dataset. Perhaps conceptually the simplest method for solving (1) is *gradient descent (GD)*,

$$x_{t+1} = x_t - \gamma \nabla \phi(x_t) = x_t - \gamma \sum_{i=1}^M \frac{n_i}{n} \nabla \phi_i(x_t) = \sum_{i=1}^M \frac{n_i}{n} \left(x_t - \gamma \nabla \phi_i(x_t) \right), \tag{2}$$

where $\gamma > 0$ is the stepsize. It will be useful to describe how GD would be implemented in a federated environment. First, all clients $i \in \{1, \ldots, M\}$ in parallel perform a single local gradient step starting from the current global model x_t, arriving at the local models $x_{it} := x_t - \gamma \nabla \phi_i(x_t)$, $i \in \{1, \ldots, M\}$. These local models are then communicated to the *orchestrating server*, which aggregates them via weighted averaging, arriving at the new global model $x_{t+1} = \sum_{i=1}^M \frac{n_i}{n} x_{it}$. This new model is then broadcast back to all clients, and the process is repeated until a model of sufficient quality is found.

1.2 Federated averaging

Proposed by Povey et al. [2015], Moritz et al. [2016], McMahan et al. [2017], federated averaging (FedAvg) is arguably the most popular method for solving the standard FL formulation (1). Motivated by the specific constraints of federated environments, FedAvg can be seen as a practical enhancement of GD via the simultaneous application of three techniques: a) data sampling (DS), b) client sampling (CS), and c) local training (LT). That is,

FedAvg = GD + (DS + CS + LT).

We will now briefly describe each of these three GD-enhancing techniques separately.

(a) GD **+ Data Sampling.** In situations when the local datasets are so large that the computation of the exact local gradients becomes a bottleneck, it makes sense to approximate them via data sampling. That is, instead of passing through all local data to compute the local gradient $\nabla \phi_i(x_t)$, each client i computes the gradients $\nabla \phi_{ij}(x_t)$ for $j \in \mathcal{D}_{it}$ only, where $\mathcal{D}_{it}$ is a suitably chosen small-enough subset of the local dataset $\{1, \ldots, n_i\}$. These gradients are then used to form gradient estimators $g_i(x_t) \approx \nabla \phi_i(x_t)$ which are used to perform a local SGD step on all clients. The rest of the procedure is the same as in the case of GD. That is, the local models obtained in this way are sent to the orchestrating server, the server aggregates them via weighted averaging and broadcasts the resulting model back to all clients. Combination of GD and DS can be seen as a particular version of SGD, where the stochastic gradient estimator is formed from the gradients $\nabla \phi_{ij}(x_t)$ associated with the datapoints (i, j) where $j \in \cup_{i=1}^M \mathcal{D}_{it}$. While DS is still an active area of research, it has been studied for a long time, and is in general well understood [Takáč et al., 2013, Li et al., 2014, Csiba and Richtárik, 2018, Gower et al., 2019a, Horváth and Richtárik, 2019, Khaled and Richtárik, 2020].

(b) GD **+ Client Sampling.** In practical federated environments, and especially in cross-device FL [Kairouz et al., 2019], the number of clients is enormous, they are not all available at all times, and the orchestrating server has limited compute and memory capacity. For these and other reasons, practical FL methods need to work in an environment in which a small subset $\mathcal{S}_t \subseteq \{1, \ldots, M\}$ of the clients is sampled ("participates") in each communication/aggregation/training round only. Since only the participating clients $i \in \mathcal{S}_t$ perform a local GD step and communicate the resulting local model to the orchestrating server for aggregation, this induces an error compared to GD, which has an adverse effect on the convergence rate. Combination of GD and CS can be seen as a particular version of SGD, where the stochastic gradient estimator is formed from the gradients $\nabla \phi_{ij}(x_t)$ associated with the datapoints (i, j) where $j \in \cup_{i=1}^{\mathcal{S}_t} \{1, \ldots, n_i\}$. While CS is still an active area of research, since CS is a special type of DS, much was known about CS long before FedAvg was proposed [Gower et al., 2019a, Horváth and Richtárik, 2019]. Still, CS poses new challenges tackled by the community Eichner et al. [2019], Chen et al. [2020], Gower et al. [2019a], Cho et al. [2020], Charles et al. [2021].

(c) GD **+ Local Training.** In federated learning, the cost of communication between the clients and the orchestrating server forms the key bottleneck. Indeed, in their FedAvg paper, which introduced LT to the world of federated learning, McMahan et al. [2017] wrote:

Table 1: Five generations of local training (LT) methods summarizing the progress made by the ML/FL community over the span of 7+ years in the understanding of the *communication acceleration properties of LT*.

Generation[(a)]	Theory	Assumptions	Comm. Complexity[(b)]	Selected Key References
1. Heuristic	✗	—	empirical results only	LocalSGD [Povey et al., 2015]
	✗	—	empirical results only	SparkNet [Moritz et al., 2016]
	✗	—	empirical results only	FedAvg [McMahan et al., 2017]
2. Homogeneous	✓	bounded gradients	sublinear	FedAvg [Li et al., 2020b]
	✓	bounded grad. diversity[(c)]	linear but worse than GD	LFGD [Haddadpour and Mahdavi, 2019]
3. Sublinear	✓	standard[(d)]	sublinear	LGD [Khaled et al., 2019]
	✓	standard	sublinear	LSGD [Khaled et al., 2020]
4. Linear	✓	standard	linear but worse than GD	Scaffold [Karimireddy et al., 2020]
	✓	standard	linear but worse than GD	S-Local-GD [Gorbunov et al., 2020a]
	✓	standard	linear but worse than GD	FedLin [Mitra et al., 2021]
5. Accelerated	✓	standard	linear & better than GD	ProxSkip/Scaffnew [Mishchenko et al., 2022]
	✓	standard	linear & better than GD	ProxSkip-VR **[THIS WORK]**

[(a)] Since client sampling (CS) and data sampling (DS) can only *worsen* theoretical communication complexity, our historical breakdown of the literature into 5 generations of LT methods focuses on the full client participation (i.e., no CS) and exact local gradient (i.e., no DS) setting. While some of the referenced methods incorporate CS and DS techniques, these are irrelevant for our purposes. Indeed, from the viewpoint of communication complexity, all these algorithms enjoy best theoretical performance in the no-CS and no-DS regime.

[(b)] For the purposes of this table, we consider problem (1) in the *smooth* and *strongly convex* regime only. This is because the literature on LT methods struggles to understand even in this simplest (from the point of view of optimization) regime.

[(c)] *Bounded gradient diversity* is a uniform bound on a specific notion of gradient variance depending on client sampling probabilities. However, this assumption (as all homogeneity assumptions) is very restrictive. For example, it is not satisfied the standard class of smooth and strongly convex functions.

[(d)] The notorious FL challenge of handling non-i.i.d. data by LT methods was solved by Khaled et al. [2019] (from the viewpoint of *optimization*). From generation 3 onwards, there was no need to invoke any data/gradient homogeneity assumptions. Handling non-i.i.d. data remains a challenge from the point of view of *generalization*, typically by considering *personalized* FL models.

> *"In contrast[1], in federated optimization communication costs dominate".*

LT is a conceptually simple and surprisingly powerful communication-acceleration technique. The basic idea behind LT is for the clients to perform *multiple* local GD steps instead of a single step (which is how GD operates) before communication and aggregation takes place. The intuitive reasoning used in virtually all papers on this topic is: performing multiple local GD steps results in "richer" and ultimately more useful local training in the sense that fewer communication rounds will *hopefully* suffice to finish the training. McMahan et al. [2017] supported this intuition with ample empirical evidence, and credited LT as the critical component behind the success of FedAvg:

> *"Thus, our goal is to use additional computation in order to decrease the number of rounds of communication needed to train a model…" "Communication costs are the principal constraint, and we show a reduction in required communication rounds by 10–100× as compared to synchronized stochastic gradient descent." "…the speedups we achieve are due primarily to adding more computation on each client".*

2 Five Generations of Local Training Methods

We now offer several historical comments on the most important developments related to the *theoretical* understanding of LT. To this end, we have identified 5 distinct generations of LT methods, each with its unique challenges and characteristics. To make the narrative simple, and since we focus on this regime in our paper, we limit our overview to loss functions ϕ_i that are μ-strongly convex and L-smooth. This is arguably the most studied class of functions in continuous optimization [Nesterov, 2004], and for this reason, it presents a valuable litmus test for any theory of LT.

2.1 Generation 1: Heuristic Age

While LT ideas were used in several machine learning domains before [Povey et al., 2015, Moritz et al., 2016], LT truly rose to prominence as a practically potent communication acceleration technique due to the seminal paper of McMahan et al. [2017] which introduced the FedAvg algorithm. However, no theory was provided in their work, nor in any prior work. LT-based heuristics, i.e., methods without any theoretical guarantees, dominated the initial development of the field up to, and including, the FedAvg paper.

[1] to datacenter optimization

2.2 Generation 2: Homogeneous Age

The first theoretical results for LT methods offering explicit convergence rates relied on various data/gradient *homogeneity*[2] assumptions. The intuitive rationale behind such assumptions comes from the following thought process. In the extreme case when all the local functions ϕ_i are *identical* (this is often referred to as the *homogeneous* or *i.i.d. data* regime), there is a very simple approach to making GD communication-efficient: push the idea of LT to its extreme by running GD on all clients, independently and in parallel, without any communication/synchronization/averaging whatsoever. Extrapolating from this, it is reasonable to assume that as we increase heterogeneity, taking multiple local steps should still be beneficial as long as we do not take too many steps. Several authors analyzed various LT methods under such assumptions, and obtained rates [Haddadpour and Mahdavi, 2019, Yu et al., 2019, Li et al., 2019, 2020b]. However, bounded dissimilarity assumptions are highly problematic. First, they do not seem to be satisfied even for some of the simplest function classes, such as strongly convex quadratics [Khaled et al., 2019, 2020], and moreover, it is well known that practical FL datasets are highly heterogeneous/non-i.i.d. McMahan et al. [2017], Kairouz et al. [2019]. So, analyses relying on such strong assumptions are both mathematically questionable, and practically irrelevant.

2.3 Generation 3: Sublinear Age

The third generation of LT methods is characterized by the successful removal of the bounded dissimilarity assumptions from the convergence theory. Khaled et al. [2019] first achieved this breakthrough by studying the simplest LT method: local gradient descent (LGD) (i.e., a simple combination of GD and LT). While works belonging to this generation elevated LT to the same theoretical footing as GD in terms of the assumptions, which marked an important milestone in our understanding of LT, unfortunately, the obtained communication complexity theory of LGD is pessimistic when compared to vanilla GD. Indeed, the inclusion of LT did *not* lead to an improvement upon the communication complexity of vanilla GD. Moreover, while GD enjoys a linear communication complexity (in the smooth and strongly convex regime), the communication complexity of LGD is *sublinear*. In a follow-up work, Khaled et al. [2020] later analyzed LGD in combination with DS as well. Woodworth et al. [2020] and Glasgow et al. [2022] provided lower bounds for LGD with DS showing that it is not better than minibatch SGD in heterogeneous setting. See the work of Malinovsky et al. [2020] for a fixed-point theory viewpoint.

2.4 Generation 4: Linear Age

The fourth generation of LT methods is characterized by the effort to design *linearly* converging variants of LT algorithms. In order to achieve this, it was important to tame the adverse effect of the so-called *client drift* [Karimireddy et al., 2020], which was identified as the culprit of the worse-than-GD theoretical performance of the previous generation of LT methods. The first LT-based method that successfully tamed client drift, and as a result obtained a linear convergence rate, was Scaffold [Karimireddy et al., 2020]. Several alternative approaches to obtaining the same effect were later proposed by Gorbunov et al. [2020a] and Mitra et al. [2021] . While obtaining a linear rate for LT methods under standard assumptions was a major achievement, the communication complexity of these methods is still somewhat worse[3] than that of vanilla GD, and is at best equal to that of GD.

2.5 Generation 5: Accelerated Age

Finally, the fifth generation of LT methods was initiated recently by Mishchenko et al. [2022] with their ProxSkip method which enjoys *accelerated communication complexity*. Acceleration comes from the LT steps coupled with a new client drift reduction technique and a probabilistic approach to deciding whether communication takes place or not. Mishchenko et al. [2022] first reformulate (1)

[2] We use the term *homogeneity* to refer to various related assumptions used in the literature, including *bounded gradient norms* [Li et al., 2020b], *bounded gradient variance* [Li et al., 2019, Yu et al., 2019] and *bounded gradient diversity* [Haddadpour and Mahdavi, 2019].

[3] Both GD, and LT methods such as Scaffold [Karimireddy et al., 2020], S-Local-GD [Gorbunov et al., 2020a] and FedLin [Mitra et al., 2021] enjoy the linear rate $\mathcal{O}(\kappa \log \frac{1}{\varepsilon})$, where κ is a condition number. However, this condition number is in general slightly worse for the LT methods.

Algorithm 1 ProxSkip-VR

Parameters: stepsize $\gamma > 0$, probability $p \in (0,1]$, initial iterate $x_0 \in \mathbb{R}^d$, initial control vector $y_0 \in \mathbb{R}^d$, initial gradient shift $h_0 \in \mathbb{R}^d$, number of iterations $T \geq 1$
for $t = 0, 1, \ldots, T-1$ **do**
  $g_t = g(x_t, y_t, \xi_t)$ ◇ Sample ξ_t and construct an unbiased estimator of $\nabla f(x_t)$
  $\hat{x}_{t+1} = x_t - \gamma(g_t - h_t)$ ◇ Take a gradient-type step adjusted via the shift h_t
  Construct new control vector y_{t+1}
  Flip a coin $\theta_t \in \{0,1\}$ where $\mathrm{Prob}(\theta_t = 1) = p$ ◇ Decides whether to skip the prox or not
  if $\theta_t = 1$ **then**
    $x_{t+1} = \mathrm{prox}_{\frac{\gamma}{p}r}\left(\hat{x}_{t+1} - \frac{\gamma}{p}h_t\right)$ ◇ Apply prox, but only with probability p
  else
    $x_{t+1} = \hat{x}_{t+1}$ ◇ Skip the prox!
  end if
  $h_{t+1} = h_t + \frac{p}{\gamma}(x_{t+1} - \hat{x}_{t+1})$ ◇ Update the shift h_t
end for

into the equivalent consensus form

$$\min_{x \in \mathbb{R}^d} f(x) + r(x), \tag{3}$$

where $d = Md'$, $x = (x_1, \ldots, x_M) \in \mathbb{R}^d$, and

$$f(x) := \sum_{i=1}^{M} \frac{n}{n_i}\phi_i(x_i), \qquad r(x) = \begin{cases} 0 & \text{if } x_1 = \cdots = x_M, \\ +\infty & \text{otherwise.} \end{cases} \tag{4}$$

The ProxSkip method is a randomized variant of proximal gradient descent (ProxGD) [Nesterov, 2013, Beck, 2017] for solving (3), with the proximity operator of r, given by

$$\mathrm{prox}_r(x) := \arg\min_y \left(r(y) + \frac{1}{2}\|y - x\|^2 \right),$$

being evaluated in each iteration with probability $p \in (0,1]$ only. Remarkably, Mishchenko et al. [2022] showed that it is possible to choose p as low as $1/\sqrt{\kappa}$, where κ is the condition number of f, without this worsening the rate of its parent method ProxGD. In summary, ProxSkip lets the M clients perform $\sqrt{\kappa}$ local gradient steps in expectation, followed by the evaluation of the prox of r, which in the case of the consensus reformulation of (1) means averaging across all M nodes.

3 ProxSkip-VR: A General Variance Reduction Framework for ProxSkip

In this work we contribute to the fifth generation of LT methods by extending the work of Mishchenko et al. [2022] to allow for a very large family of gradient estimators, including variance reduced (VR) ones [Johnson and Zhang, 2013, Defazio et al., 2014, Kovalev et al., 2020a, Mishchenko et al., 2019].

Like ProxSkip, our method ProxSkip-VR (Algorithm 1) is aimed to solve the composite problem (3) in a more general setting (see Assumptions 1–3), with the special structure (4) coming from the consensus reformulation being a special case only. Our method differs from ProxSkip in that we replace the gradient $\nabla f(x_t)$ by an unbiased estimator $g_t = g(x_t, y_t, \xi_t)$, where ξ_t is the source of randomness controlling unbiasedness and y_t is a control vector whose role is to progressively reduce the variance of the estimator, so that

$$\mathbb{E}\left[g_t \mid x_t, y_t\right] = \nabla f(x_t).$$

There are several motivations behind this endeavor. First, it is a-priori not clear whether the novel proof technique employed by Mishchenko et al. [2022] can be combined with the proof techniques used in the analysis of VR methods, and hence it is scientifically significant to investigate the possibility of such a merger of two strands of the literature. We show that this is possible. Second, marrying VR estimators with ProxSkip can lead to novel system architectures which are more elaborate than the simplistic client-server architecture (see Section 4). Lastly, while researchers contributing to generations 1–4 of LT methods were preoccupied with trying to close the gap on GD

in terms of communication efficiency, they *ignored* the number of the local steps appearing in their algorithms, and reported their bounds primarily in terms of the number of communication rounds. Bounds reported this way make complete sense in the scenario when the cost of local work (e.g., one SGD step w.r.t. a single data point), say δ, is negligible compared to the cost of communication, which can w.l.o.g. assume to be 1, and when the number of local steps is small. With the advent of the fifth generation of LT methods, we can (to a large degree) stop worrying about communication efficiency, and can now ask more refined questions, such as:

> *Are there gradient estimators which, when combined with ProxSkip, lead to faster algorithms in terms of the total cost, which includes the communication cost as well as the cost of local training?*

We give an affirmative answer to the question in Sections 4 and 5.

3.1 Standard assumptions

We assume throughout that f is differentiable, and let $D_f(x,y) := f(x) - f(y) - \langle \nabla f(y), x-y\rangle$ denote the Bregman divergence of f. Throughout the work we make the following assumptions:

Assumption 1 (L-smoothness)**.** *$\exists L > 0$ such that $2D_f(x,y) \leq L\left\|x-y\right\|^2$ for all $x,y \in \mathbb{R}^d$.*

Assumption 2 (μ-convexity)**.** *$\exists \mu > 0$ such that $\mu\left\|x-y\right\|^2 \leq 2D_f(x,y)$ for all $x,y \in \mathbb{R}^d$.*

Assumption 3. *The regularizer $r : \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ is proper, closed and convex.*

Under the above assumptions, (3) has a unique minimizer $x_\star$. Let $h_\star := \nabla f(x_\star)$.

3.2 Modelling variance reduced gradient estimators

Our next assumption, initially introduced by Gorbunov et al. [2020b], postulates several parametric inequalities characterizing the behavior and ultimately the quality of a gradient estimator. Similar assumptions appeared later in [Gorbunov et al., 2020a,c].

Assumption 4. *Let $\{x_t\}$ be iterates produced by ProxSkip-VR. First, we assume that the stochastic gradients $g_t = g(x_t, y_t, \xi_t)$ are unbiased for all $t \geq 0$, namely*

$$\mathbb{E}\left[g_t \mid x_t, y_t\right] = \nabla f(x_t). \tag{5}$$

Second, we assume that there exist non-negative constants $A, B, C, \tilde{A}, \tilde{B}, \tilde{C}$, with $\tilde{B} < 1$, and a nonnegative mapping $y_t \mapsto \sigma(y_t) := \sigma_t$ such that the following two relations hold for all $t \geq 0$,

$$\mathbb{E}\left[\left\|g_t - \nabla f(x_*)\right\|^2 \mid x_t, y_t\right] \leq 2AD_f(x_t, x_*) + B\sigma_t + C, \tag{6}$$

$$\mathbb{E}\left[\sigma_{t+1} \mid x_t, y_t\right] \leq 2\tilde{A}D_f(x_t, x_*) + \tilde{B}\sigma_t + \tilde{C}. \tag{7}$$

Assumption 4 covers a very large collection of gradient estimators, including an infinite variety of subsampling/minibatch estimators, gradient sparsification and quantization estimators, and their combinations; see [Gorbunov et al., 2020b] for examples. VR estimators are characterized by $C = \tilde{C} = 0$; most non-VR estimators by $\tilde{A} = \tilde{B} = \tilde{C} = B = 0$ and $C > 0$ [Gower et al., 2019b].

3.3 Main result

We are now ready to formulate our main result.

Theorem 5. *Let Assumptions 2 and 3 hold, and let g_t be a gradient estimator satisfying Assumption 4. If $B > 0$, choose any $W > B/(1-\tilde{B})$ and $\beta = (B + W\tilde{B})/W$. If $B = 0$, let $W = 0$ and $\beta = \tilde{B}$. Choose stepsize $0 < \gamma \leq \min\left\{1/\mu, 1/(A + W\tilde{A})\right\}$. Then the iterates of ProxSkip-VR for any $p \in (0,1]$ satisfy*

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, \beta^T, (1-p^2)^T\right\}\Psi_0 + \frac{\left(C + W\tilde{C}\right)\gamma^2}{\min\left\{\gamma\mu, p^2, 1-\beta\right\}}, \tag{8}$$

where the Lyapunov function is defined by $\Psi_t := \left\|x_t - x_\star\right\|^2 + \frac{\gamma^2}{p^2}\left\|h_t - h_\star\right\|^2 + \gamma^2 W\sigma_t$.

Table 2: Special cases of ProxSkip-VR, depending on the choice of the gradient estimator g_t.

Estimator of ∇f	Communication Complexity of ProxSkip-VR	Iteration Complexity of ProxSkip-VR	Corollaries of Theorem 5
GD [(b)]	$\mathcal{O}\left(\sqrt{L/\mu}\log 1/\varepsilon\right)$	$\mathcal{O}\left(L/\mu \log 1/\varepsilon\right)$	Theorem 6
SGD [(c)]	$\mathcal{O}\left(\left(\sqrt{A/\mu}+\sqrt{2C/\varepsilon\mu^2}\right)\log 1/\varepsilon\right)$	$\mathcal{O}\left(\left(A/\mu + 2C/\varepsilon\mu^2\right)\log 1/\varepsilon\right)$	Theorem 8
HUB **[NEW]**	$\mathcal{O}\left(\sqrt{L_{\max}/\mu\,(1+\omega/\tau)}\log 1/\varepsilon\right)$	$\mathcal{O}\left(L_{\max}/\mu\,(1+\omega/\tau)\log 1/\varepsilon\right)$	Theorem 9
LSVRG **[NEW]**	$\mathcal{O}\left(\sqrt{L(\tau)/\mu}\log 1/\varepsilon\right)$	$\mathcal{O}\left(L(\tau)/\mu \log 1/\varepsilon\right)$	Corollary 1
Q **[NEW]**	$\mathcal{O}\left(\sqrt{L_{\max}/\mu\,(1+\omega/M)}\log 1/\varepsilon\right)$	$\mathcal{O}\left(L_{\max}/\mu\,(1+\omega/M)\log 1/\varepsilon\right)$	Corollary 2

[(a)] Any estimator satisfying Assumption 4
[(b)] ProxSkip-VR with the GD estimator reduces to the ProxSkip method of Mishchenko et al. [2022]
[(c)] ProxSkip-VR with the SGD estimator satisfying Assumption 7 reduces to the SProxSkip method of Mishchenko et al. [2022]
[(d)] $L(\tau) := \frac{m-\tau}{\tau(m-1)}L_{\max} + \frac{m(\tau-1)}{\tau(m-1)}L$, where τ is the mini-batch size and m is the number of clients belonging to one hub

3.4 Two examples of gradient estimators

Here we give two illustrating examples of estimators satisfying Assumption 4.

Theorem 6 (GD estimator)**.** *Let Assumption 1, 2 and 3 hold. Then for the trivial estimator $g_t = \nabla f(x_t)$, Assumption 4 holds with the following parameters:*

$$A = L, \quad B = 0, \quad C = 0, \quad \tilde{A} = 0, \quad \tilde{B} = 0, \quad \tilde{C} = 0, \quad \sigma_t \equiv 0.$$

Choose a stepsize satisfying $0 < \gamma \leq 1/L$. Then the iterates of ProxSkip-VR for any $p \in (0,1]$ satisfy

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T\right\}\Psi_0, \tag{9}$$

where $\Psi_t := \|x_t - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2$. Let $\gamma = 1/L$ and $p = \sqrt{\mu/L}$ then the communication and iteration complexities of ProxSkip-VR are $\#comms = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{1}{\varepsilon}\right), \#iters = \mathcal{O}\left(\frac{L}{\mu}\log\frac{1}{\varepsilon}\right)$.

This recovers the result obtained by Mishchenko et al. [2022] for their ProxSkip method. The next assumption holds for virtually all (non-VR) estimators based on subsampling [Gower et al., 2019b].

Assumption 7 (Expected smoothness)**.** *We say that an unbiased estimator $g(x;\xi) : \mathbb{R}^d \to \mathbb{R}^d$ of the gradient $\nabla f(x)$ satisfies the expected smoothness inequality if there exists $A'' > 0$ such that*

$$\mathbb{E}\left[\|g(x;\xi) - g(x_\star;\xi)\|^2\right] \leq 2A''D_f(x, x_\star), \quad \forall x \in \mathbb{R}^d.$$

Theorem 8 (SGD estimator)**.** *Let $g(x,\xi)$ satisfy Assumption 7 and define $g_t := g(x_t, \xi_t)$, where ξ_t is chosen independently at time t. Then Assumption 4 holds with the following parameters:*

$$A = 2A'', \quad B = 0, \quad C = 2\mathrm{Var}(g(x_\star,\xi)), \quad \tilde{A} = 0, \quad \tilde{B} = 0, \quad \tilde{C} = 0, \quad \sigma_t \equiv 0.$$

Moreover, assume that Assumption 2 holds. Choose stepsize $0 < \gamma \leq \min\left\{1/\mu, 1/A\right\}$. Then the iterates of ProxSkip-VR for any probability $p \in (0,1]$ satisfy

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T\right\}\Psi_0 + \gamma^2\frac{2\mathrm{Var}(g(x_\star,\xi))}{\min\left\{\gamma\mu, p^2\right\}}, \tag{10}$$

where the Lyapunov function is defined by $\Psi_t := \|x_t - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2$. If we choose $\gamma = \min\left\{1/A, \varepsilon\mu/2C\right\}$ and $p = \sqrt{\gamma\mu}$ then the communication and iteration complexities of ProxSkip-VR are $\#comms = \mathcal{O}\left(\left(\sqrt{\frac{A}{\mu}} + \sqrt{\frac{C}{\varepsilon\mu^2}}\right)\log\frac{1}{\varepsilon}\right), \#iters = \mathcal{O}\left(\left(\frac{A}{\mu} + \frac{C}{\varepsilon\mu^2}\right)\log\frac{1}{\varepsilon}\right)$, respectively.

This recovers the result obtained by Mishchenko et al. [2022] for their stochastic variant of ProxSkip, which they call SProxSkip.

4 New FL Architecture: Regional Hubs Connecting the Clients to the Server

We now illustrate the versatility of our ProxSkip-VR framework by designing a new "FL architecture" and proposing an algorithm that can efficiently operate in this setting.

In particular, we consider the situation where the clients are clustered (e.g., based on region), and where *a hub* is placed in between each cluster and the central server. Clients communicate with their regional hub only, which can communicate with the central server (see Figure 1). There are M hubs, hub i handles n_i clients, and client j associated with hub i owns loss function ϕ_{ij}.

Mathematically, this can be modeled by problem (1). In this situation, we care about two sources of communication cost: the server and the hubs, and between the hubs and the clients. We propose to handle this via *local training (LT)* between the server and the hubs, and via *client sampling (CS)* and *compressed communication (CC)* between the hubs and the clients. Algorithmically, from the server-hubs perspective, we are applying a particular variant of ProxSkip-VR to (3)–(4), where ϕ_i is the aggregate loss handles by hub i. This takes care of communication efficiency between the server and the hub. Note also that we need not worry about partial participation of hubs, as these are designed to be always available. However, in this situation, it is costly for hub i to compute the gradient of ϕ_i as this involves communication with all the clients it handles.

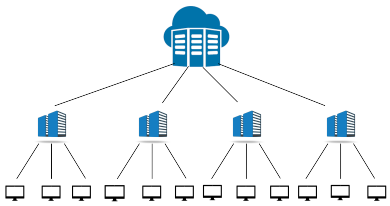


Figure 1: Server-hubs-clients FL architecture with 4 hubs and 12 clients.

4.1 Handling of client sampling (CS) and compressed communication (CC)

In order to alleviate this burden, we propose a combination of CS and CC. However, we need to be very careful about how to do this. Indeed, both CS and CC, even when applied in isolation, and without ProxSkip in the mix, can lead to a substantial slowdown in convergence. For example, one will typically lose linear convergence in the strongly convex regime. However, techniques for preserving linear convergence in the presence of CS and CC exist: this is what variance reduction strategies are designed to do. For example, LSVRG [Hofmann et al., 2015, Kovalev et al., 2020a] is a VR technique for reducing the variance due to CS, and DIANA [Mishchenko et al., 2019] is a VR technique for reducing the variance due to CC. However, we are not aware of any VR method that combines CS (applied first) and CC (applied second).

We now propose such a technique. In iteration t, every hub $i \in \{1, 2, \ldots, M\}$ selects a random subset $\mathcal{S}_t^i \subseteq \{1, 2, \ldots, n_i\}$ of the clients it handles of cardinality τ_i, chosen uniformly at random, and estimates the hub gradient via

$$\nabla \phi_i(x_t) \approx g_t^i := \frac{1}{|\mathcal{S}_t^i|} \sum_{j \in \mathcal{S}_t^i} \mathcal{Q}_t^{ij} \left(\nabla \phi_{ij}(x_t) - \nabla \phi_{ij}(y_t) \right) + \nabla \phi_i(y_t), \tag{11}$$

where $\mathcal{Q}_t^{ij} : \mathbb{R}^{d'} \to \mathbb{R}^{d'}$ is a randomized compression (e.g., sparsification or quantization) operator [Alistarh et al., 2017, Khirirat et al., 2018, Horváth et al., 2019b,a, Philippenko and Dieuleveut, 2020], i.e., a mapping satisfying

$$\mathbb{E}\left[Q_t^{ij}(x) \right] = x, \quad \mathbb{E}\left[\|Q_t^{ij}(x) - x\|^2 \right] \leq \omega \|x\|^2, \quad \forall x \in \mathbb{R}^{d'}, \tag{12}$$

and the control vector y_t is updated probabilistically as follows:

$$y_{t+1} = \begin{cases} x_t & \text{with probability} \quad q \\ y_t & \text{with probability} \quad 1-q \end{cases}. \tag{13}$$

The global gradient estimator (a vector in $\mathbb{R}^{Md'}$), which we call HUB, is construcuted as a concatenation of the above hub estimators:

$$\nabla f(x_t) := \left(\frac{n_i}{n} \nabla \phi_i(x_t) \right)_{i=1}^M \approx g_t := g(x_t, y_t, \xi_t) := \left(\frac{n_i}{n} g_t^i \right)_{i=1}^M, \tag{14}$$

where ξ_t represents the combined randomness from the compressors $\{\mathcal{Q}_t^{ij}\}$ and random sets $\{\mathcal{S}_t^i\}$.

In order to analyze ProxSkip-VR in the consensus form, from now on we assume that $n_i = m = {}^n\!/_M$ and $\tau_i = \tau \in \{1, 2, \ldots, m\}$ for all i, and rely on a slightly different, more general reformulation:

$$\min_{x \in \mathbb{R}^d} \frac{1}{m} \sum_{j=1}^m \widetilde{\phi}_j(x) + r(x), \quad \widetilde{\phi}_j(x) := \frac{1}{M} \sum_{i=1}^M \phi_{ij}(x_i), \quad r(x) := \begin{cases} 0 & \text{if } x_1 = \cdots = x_M, \\ +\infty & \text{otherwise.} \end{cases}$$

Our proposed method ProxSkip-HUB is ProxSkip-VR combined with the novel HUB estimator (14), applied to the above reformulation; see Algorithm 2.

4.2 Theory for ProxSkip-HUB

In the following result we first claim that the above estimator satisfies Assumption 4 with $C = \tilde{C} = 0$ (i.e., it is variance-reduced), and the rest of the claim follows by application of our general theorem Theorem 5.

Theorem 9. *Assume that $\nabla\widetilde{\phi}_j$ is L_j-smooth for all j and let Assumptions 2 and 3 hold. Then for the gradient estimator (14), Assumption 4 holds with the following constants:*

$$A = 4\left(L(\tau) + \frac{\omega}{\tau}L_{\max}\right), \quad B = 4\left(1 + \frac{\omega}{\tau}\right), \quad C = 0, \quad \tilde{A} = qL_{\max}, \quad \tilde{B} = 1 - q, \quad \tilde{C} = 0,$$

and $\sigma_t := \sigma(y_t)$, $\sigma(y) := \frac{1}{m}\sum_{j=1}^m \|\nabla\widetilde{\phi}_j(y) - \nabla\widetilde{\phi}_j(x_\star)\|^2$, $L_{\max} := \max_j L_j$. Set $W = {}^{2B}/_{(1-\tilde{B})}$ and $0 < \gamma \leq \min\left\{{}^1/_\mu, {}^1/_{(A+W\tilde{A})}\right\}$. Then the iterates of ProxSkip-VR *for any $p \in (0,1]$ satisfy*

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T, \left(1 - \frac{q}{2}\right)^T\right\}\Psi_0,$$

where the Lyapunov function is defined by $\Psi_t := \|x_t - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2 + \gamma^2\frac{8}{q}\left(1 + \frac{\omega}{\tau}\right)\sigma_t$.

We now consider two special cases. In the first, we specialize to the no compression regime, and in the second, to the full participation regime.

Corollary 1 (No compression)**.** *If we do not use compression (i.e., $\omega = 0$), then the communication and iteration complexities are $\#comms = \mathcal{O}\left(\sqrt{\frac{L_{\max}}{\mu}}\log\frac{1}{\varepsilon}\right)$, $\#iters = \mathcal{O}\left(\frac{L_{\max}}{\mu}\log\frac{1}{\varepsilon}\right)$, respectively. However, if we use the estimator (11) in Theorem 5 directly, then the communication and iteration complexities are $\#comms = \mathcal{O}\left(\sqrt{\frac{L(\tau)}{\mu}}\log\frac{1}{\varepsilon}\right)$, $\#iters = \mathcal{O}\left(\frac{L(\tau)}{\mu}\log\frac{1}{\varepsilon}\right)$, where $L(\tau) := \frac{m-\tau}{\tau(m-1)}L_{\max} + \frac{m(\tau-1)}{\tau(m-1)}L$.*

Notice that $L_{\max} \geq L$, and that $L(\tau) = L_{\max}$ for $\tau = 1$ and $L(\tau) = L$ for $\tau = m$. Moreover, $L(\tau)$ decreases as the minibatch size τ increases.

Corollary 2 (No client sampling)**.** *If we do not use client sampling (i.e., $\tau = m$), then the communication and iteration complexities are $\#comms = \mathcal{O}\left(\sqrt{\frac{L_{\max}}{\mu}\left(1 + \frac{\omega}{M}\right)}\log\frac{1}{\varepsilon}\right)$, $\#iters = \mathcal{O}\left(\frac{L_{\max}}{\mu}\left(1 + \frac{\omega}{M}\right)\log\frac{1}{\varepsilon}\right)$, respectively.*

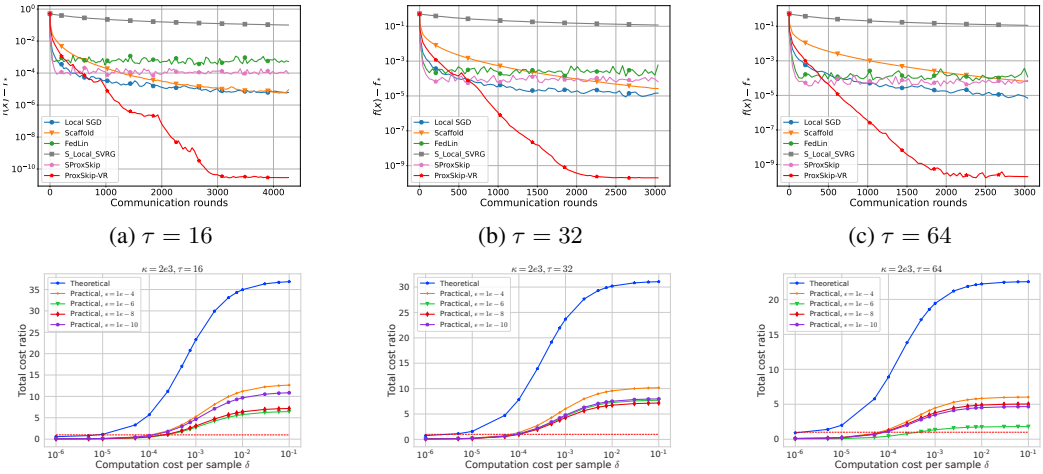
Assume $r(x) \equiv 0$. If $\mathcal{Q}_t^{ij}(x) \equiv x$, then $\omega = 0$, and we restore the well-known LSVRG method [Hofmann et al., 2015, Kovalev et al., 2020b], assuming that the functions ϕ_{ij} have the same smoothness constant. We can recover the same rate exactly as well, but with a slightly more refined analysis, one in which we do not need to work with compressors (LSVRG does not involve any), which makes for a tighter analysis. If $\tau = m$, we restore the rate of the well-known DIANA [Mishchenko et al., 2019, Horváth et al., 2019b] method and its sibling Rand-DIANA [Shulgin and Richtárik, 2021].

5 Experiments

To illustrate the predictive power of our theory, it suffices to consider L_2-regularized logistic regression in the distributed setting (1), with

$$\phi_i(x) = \frac{1}{n_i}\sum_{j=1}^{n_i}\log\left(1 + \exp\left(-b_{ij}a_{ij}^\top x\right)\right) + \frac{\lambda}{2}\|x\|^2,$$

where n_i is the number of data points per worker $a_{ij} \in \mathbb{R}^{d'}$ and $b_{ij} \in \{-1, +1\}$ are the data samples and labels. We choose $n_i = m = {}^n/_M$ for all i. We set the regularization parameter $\lambda = 5 \cdot 10^{-4}L$ by default, where L is the smoothness constant of f. We conduct several experiments on the `w8a` dataset from LibSVM library [Chang and Lin, 2011]. In Figure 2 (first row), we compare various LT baselines[4] for three choices of mini-batch sizes ($\tau = 16, 32, 64$) with our method ProxSkip-VR combined with the LSVRG estimator, which is a special case of the HUB estimator when



(a) $\tau = 16$ (b) $\tau = 32$ (c) $\tau = 64$

Figure 2: The top row shows the convergence results compared with baselines and the second row is the total cost ratio of ProxSkip over our ProxSkip-VR.

$\mathcal{Q}_t^{ij}(x) \equiv x$ for all i, j. We see that our method outperforms all other methods significantly due to its communication-acceleration properties.

Next, we derive the total cost, which includes communication cost (assumed to be 1, for normalization purposes), and computational cost (assumed to be δ; and equal to the cost of performing one SGD step with a single data point). Let us consider the total cost of ProxSkip-VR in the case of the LSVRG estimator: in each iteration we compute 1 stochastic gradient, and with probability q we compute the exact gradient. We do not need to compute a second stochastic gradient since we can use memory and the relation $y_{t+1} = x_t$. The total cost for ProxSkip-VR is equal to

$$\text{Cost}(\text{ProxSkip-VR}) := T_{\text{comm.}}(\text{ProxSkip-VR}) + \delta\left(qm + (1-q)\tau + \tau\right) T_{\text{iter}}(\text{ProxSkip-VR}).$$

ProxSkip requires full/exact gradient computation at each iteration, so the total cost of ProxSkip is

$$\text{Cost}(\text{ProxSkip}) := T_{\text{comm.}}(\text{ProxSkip}) + \delta m T_{\text{iter}}(\text{ProxSkip}).$$

Using Theorems 6 and 9 and the value $L(\tau) := \frac{m-\tau}{\tau(m-1)} L_{\max} + \frac{m(\tau-1)}{\tau(m-1)} L$ of the expected smoothness constant for sampling with minibatch size τ uniformly at random, we get the following expression for the cost ratio, expressed as a function of δ:

$$\text{Cost ratio}(\delta) := \frac{\text{Cost}(\text{ProxSkip})}{\text{Cost}(\text{ProxSkip-VR})} = \frac{\sqrt{\mu L} + mL\delta}{\sqrt{\mu L(\tau)} + \left(2m\mu + (2L(\tau) - 2\mu)\tau\right)\delta}. \tag{15}$$

We can easily calculate the limits of this expression:

$$\text{Cost ratio}(\delta = 0) = \sqrt{\frac{L}{L(\tau)}}, \qquad \text{Cost ratio}(\delta \to \infty) = \frac{mL}{2(m\mu + (L(\tau) - \mu)\tau)}.$$

In Figure 2 (second row), we depict the theoretical cost ratio according to (15) and the corresponding experimental cost ratio obtained by an actual run of both methods to achieve ε-accuracy, with $\varepsilon = 10^{-6}$ and $\varepsilon = 10^{-8}$. Remarkably, the experimental results follows the same pattern as our theoretical prediction. The empirical curves appear lower because we use approximations for $L_{\max}$, L and μ. As we can see, starting from $\delta = 10^{-4}$, ProxSkip-VR starts to outperform ProxSkip. As δ grows, the advantage of variance reduction embedded in ProxSkip-VR over vanilla ProxSkip grows. These results suggest that variance reduction is of practical utility in terms of the total cost, even for small values of δ, and its effectiveness grows with δ.

Acknowledgements

We would like to thank Eduard Gorbunov for useful discussions related to some aspects of the theory.

[4]With the exception of LocalSGD, all use client drift correction.

References

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

A. Beck. *First order methods in optimization*. MOS-SIAM Series on Optimization, 2017.

K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

C.-C. Chang and C.-J. Lin. LibSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

Z. Charles, Z. Garrett, Z. Huo, S. Shmulyian, and V. Smith. On large-cohort training for federated learning. *arXiv preprint arXiv:2106.07820*, 2021.

W. Chen, S. Horváth, and P. Richtárik. Optimal client sampling for federated learning. *Privacy Preserving Machine Learning (NeurIPS 2020 Workshop)*, 2020.

Y. J. Cho, J. Wang, and G. Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.

D. Csiba and P. Richtárik. Importance sampling for minibatches. *Journal of Machine Learning Research*, 19(27):1–21, 2018.

A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems 27*, 2014.

H. Eichner, T. Koren, H. B. McMahan, N. Srebro, and K. Talwar. Semi-cyclic stochastic gradient descent. In *International Conference on Machine Learning*, 2019.

M. R. Glasgow, H. Yuan, and T. Ma. Sharp bounds for federated averaging (local sgd) and continuous perspective. In *International Conference on Artificial Intelligence and Statistics*, pages 9050–9090. PMLR, 2022.

E. Gorbunov, F. Hanzely, and P. Richtárik. Local SGD: unified theory and new efficient methods. In *NeurIPS*, 2020a.

E. Gorbunov, F. Hanzely, and P. Richtárik. A unified theory of SGD: Variance reduction, sampling, quantization and coordinate descent. In *International Conference on Artificial Intelligence and Statistics*, pages 680–690. PMLR, 2020b.

E. Gorbunov, D. Kovalev, D. Makarenko, and P. Richtárik. Linearly converging error compensated SGD. *Advances in Neural Information Processing Systems*, 33:20889–20900, 2020c.

R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik. SGD: General analysis and improved rates. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5200–5209, Long Beach, California, USA, 09–15 Jun 2019a. PMLR. URL http://proceedings.mlr.press/v97/qian19b.html.

R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin, and P. Richtárik. Sgd: General analysis and improved rates. In *International Conference on Machine Learning*, pages 5200–5209. PMLR, 2019b.

F. Haddadpour and M. Mahdavi. On the convergence of local descent methods infederated learning. *arXiv preprint arXiv:1910.14425*, 2019.

T. Hofmann, A. Lucchi, S. Lacoste-Julien, and B. McWilliams. Variance reduced stochastic gradient descent with neighbors. *Advances in Neural Information Processing Systems*, 28, 2015.

S. Horváth and P. Richtárik. Nonconvex variance reduced optimization with arbitrary sampling. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2781–2789, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/horvath19a.html.

S. Horváth, C.-Y. Ho, Ľudovít Horváth, A. N. Sahu, M. Canini, and P. Richtárik. Natural compression for distributed deep learning. *arXiv preprint arXiv:1905.10988*, 2019a.

S. Horváth, D. Kovalev, K. Mishchenko, S. Stich, and P. Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*, 2019b.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems 26*, pages 315–323, 2013.

P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. D'Oliveira, H. Eichner, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konečný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, M. Raykova, H. Qi, D. Ramage, R. Raskar, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao. Advances and open problems in federated learning. *Foundations and Trends®in Machine Learning*, 14(1–2):1–210, 2019.

S. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. Suresh. SCAFFOLD: Stochastic controlled averaging for on-device federated learning. In *ICML*, 2020.

A. Khaled and P. Richtárik. Better theory for SGD in the nonconvex world. *arXiv Preprint arXiv:2002.03329*, 2020.

A. Khaled, K. Mishchenko, and P. Richtárik. First analysis of local GD on heterogeneous data. In *NeurIPS Workshop on Federated Learning for Data Privacy and Confidentiality*, pages 1–11, 2019.

A. Khaled, K. Mishchenko, and P. Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS 2020)*, 2020.

S. Khirirat, H. R. Feyzmahdavian, and M. Johansson. Distributed learning with compressed gradients. *arXiv preprint arXiv:1806.06573*, 2018.

J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik. Federated optimization: distributed machine learning for on-device intelligence. *arXiv:1610.02527*, 2016a.

J. Konečný, H. B. McMahan, F. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. Federated learning: strategies for improving communication efficiency. In *NIPS Private Multi-Party Machine Learning Workshop*, 2016b.

D. Kovalev, S. Horváth, and P. Richtárik. Don't jump through hoops and remove those loops: SVRG and Katyusha are better without the outer loop. In *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, 2020a.

D. Kovalev, S. Horváth, and P. Richtárik. Don't jump through hoops and remove those loops: Svrg and katyusha are better without the outer loop. In *Algorithmic Learning Theory*, pages 451–467. PMLR, 2020b.

M. Li, T. Zhang, Y. Chen, and A. J. Smola. Efficient mini-batch training for stochastic optimization. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 661–670, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. doi: 10.1145/2623330.2623612. URL http://doi.acm.org/10.1145/2623330.2623612.

T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020a. doi: 10.1109/MSP.2020.2975749.

X. Li, W. Yang, S. Wang, and Z. Zhang. Communication-efficient local decentralized SGD methods. *arXiv preprint arXiv:1910.09126*, 2019.

X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. On the conver-gence of FedAvg on non-IID data. In *International Conference on Learning Representations*, 2020b.

G. Malinovsky, D. Kovalev, E. Gasanov, L. Condat, and P. Richtárik. From local SGD to local fixed point methods for federated learning. In *International Conference on Machine Learning*, 2020.

B. McMahan and D. Ramage. Federated learning: Collaborative machine learning without centralized training data. GoogleAIBlog, Apr. 2017.

H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

K. Mishchenko, E. Gorbunov, M. Takáč, and P. Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019.

K. Mishchenko, G. Malinovsky, S. Stich, and P. Richtárik. ProxSkip: Yes! Local gradient steps provably lead to communication acceleration! Finally! In *39th International Conference on Machine Learning (ICML 2022)*, 2022.

A. Mitra, R. Jaafar, G. Pappas, and H. Hassani. Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients. In *Advances in Neural Information Processing Systems 34*, 2021.

P. Moritz, R. Nishihara, I. Stoica, and M. I. Jordan. SparkNet: Training deep networks in Spark. In *International Conference on Learning Representations (ICLR)*, 2016.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course (Applied Optimization)*. Kluwer Academic Publishers, 2004.

Y. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

C. Philippenko and A. Dieuleveut. Bidirectional compression in heterogeneous settings for distributed or federated learning with partial participation: tight convergence guarantees. *arXiv preprint arXiv:2006.14591*, 2020.

D. Povey, X. Zhang, and S. Khudanpur. Parallel training of DNNs with natural gradient and parameter averaging. In *ICLR Workshop*, 2015.

E. Shulgin and P. Richtárik. Shifted compression framework: Generalizations and improvements. In *OPT2021: 13th Annual Workshop on Optimization for Machine Learning*, 2021.

M. Takáč, A. Bijral, P. Richtárik, and N. Srebro. Mini-batch primal and dual methods for SVMs. In *30th International Conference on Machine Learning*, pages 537–552, 2013.

J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, B. A. y Arcas, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data, S. Diggavi, H. Eichner, A. Gadhikar, Z. Garrett, A. M. Girgis, F. Hanzely, A. Hard, C. He, S. Horvath, Z. Huo, A. Ingerman, M. Jaggi, T. Javidi, P. Kairouz, S. Kale, S. P. Karimireddy, J. Konecny, S. Koyejo, T. Li, L. Liu, M. Mohri, H. Qi, S. J. Reddi, P. Richtarik, K. Singhal, V. Smith, M. Soltanolkotabi, W. Song, A. T. Suresh, S. U. Stich, A. Talwalkar, H. Wang, B. worth, S. Wu, F. X. Yu, H. Yuan, M. Zaheer, M. Zhang, T. Zhang, C. Zheng, C. Zhu, and W. Zhu. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917*, 2021.

B. E. Woodworth, K. K. Patel, and N. Srebro. Minibatch vs local sgd for heterogeneous distributed learning. *Advances in Neural Information Processing Systems*, 33:6281–6292, 2020.

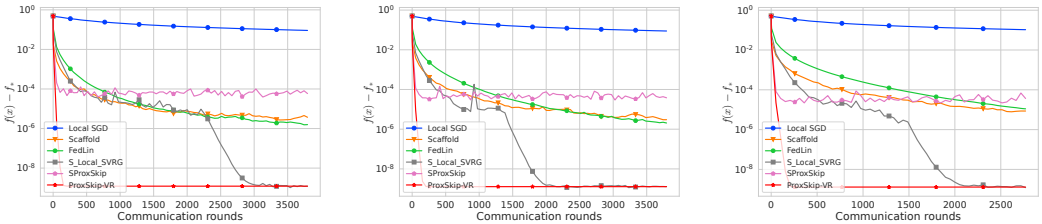
H. Yu, R. Jin, and S. Yang. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. In *International Conference on Machine Learning (ICML)*, 2019.

Appendix

A Supplementary Experiments

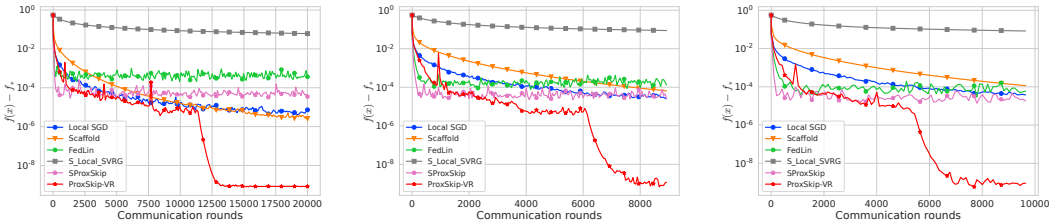
A.1 Additional experiments with ProxSkip-VR

We conduct further experiments to validate the efficiency of our proposed method ProxSkip-VR. We instantiate our variance reduction design with LSVRG and compare our method with several baselines across various datasets (w8a/a9a), different number of workers (10/20), and different batch sizes (16/32/64). All results in Figures 3, 4, 5, 6 show that ProxSkip-VR achieves linear convergence and outperforms the baselines.



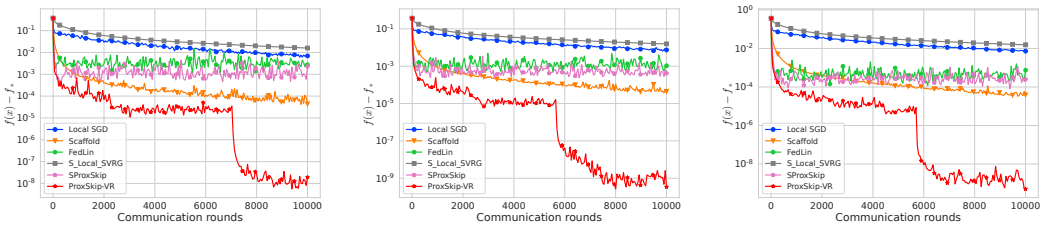
(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 3: Convergence results with 20 distributed workers on w8a dataset, $\kappa = 1e3$.



(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 4: Convergence results with 20 distributed workers on w8a dataset, $\kappa = 1e4$.

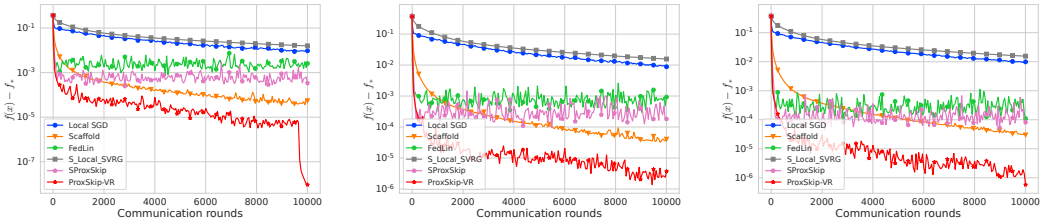


(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 5: Convergence results with 10 distributed workers on a9a dataset, $\kappa = 1e4$.

A.2 Additional experiments of cost ratio of ProxSkip over ProxSkip-VR

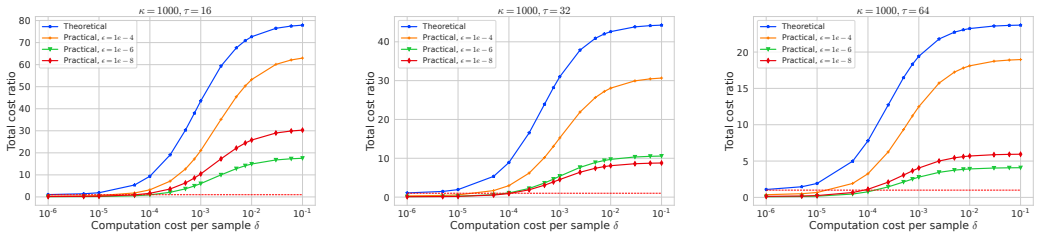
Here we report on more experiments related to the total cost ratio of ProxSkip over ProxSkip-VR in Figures 7, 8, 9, 10. When GD is used as the subroutine, i.e., when ProxSkip-VR reduces to ProxSkip, the ratio is equal to one, and is depicted by the red horizontal dashed line. Any cost ratio above one means that ProxSkip-VR benefits over ProxSkip (e.g., a cost ratio of 10 means $10\times$ speedup in favor of our method). As seen in the plots, acceleration of our method over ProxSkip is clearly visible, and improves as the local computation cost δ per sample increases. For large values of δ (i.e., $\delta \approx 10^{-1}$),



(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

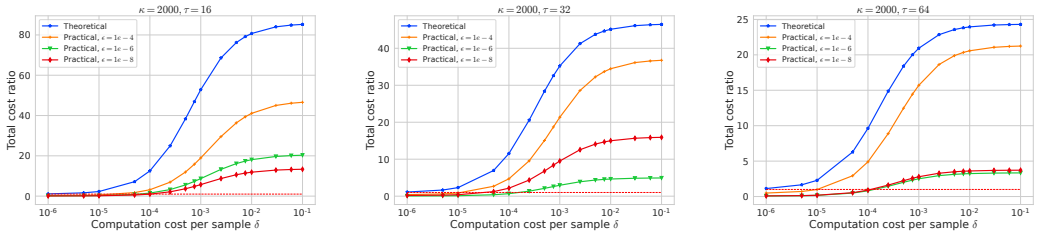
Figure 6: Convergence results with 20 distributed workers on a9a dataset, $\kappa = 1e4$.

the acceleration can reach $20\times$ to $85\times$. Note also that acceleration is more significant for small minibatch sizes. That is, it is better for $\tau = 16$ than for $\tau = 32$, which is better than in the $\tau = 64$ case. This means that in terms of total cost, it is beneficial for the workers to use smaller minibatch sizes, i.e., it is beneficial to be as far from the full batch regime employed by ProxSkip as possible.



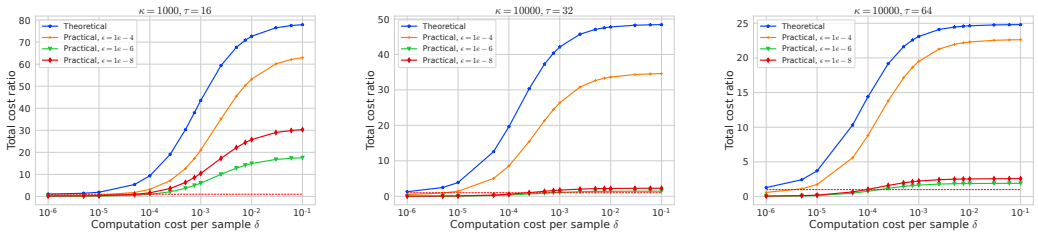
(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 7: Acceleration with 10 distributed workers on a9a dataset, $\kappa = 1e3$.



(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 8: Acceleration with 10 distributed workers on a9a dataset, $\kappa = 2e3$.

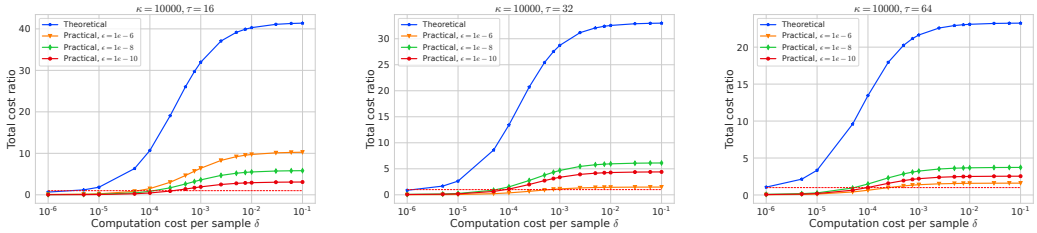


(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 9: Acceleration with 10 distributed workers on a9a dataset, $\kappa = 1e4$.

A.3 Experiments with ProxSkip-HUB

In this work we introduced a new FL architecture: regional hubs connecting the clients to the server; see Section 4. For conceptual simplicity, and in order to facilitate fair comparison with ProxSkip-



(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 10: Acceleration with 10 distributed workers on w8a dataset, $\kappa = 1e4$.

LSVRG, we assume that the number of hubs equals to the number of clients, and that each client owns a single datapoint only. We compare ProxSkip-HUB with ProxSkip-LSVRG to check whether communication compression leads to any benefits in terms of total costs. Theoretically, and similarly to our analysis in Section 5, the total cost for ProxSkip-LSVRG is

$$\text{Cost}(\text{ProxSkip-LSVRG}) := T_{\text{comm.}}(\text{ProxSkip-LSVRG}) + \delta\left(qm + (1-q)\tau + \tau\right) T_{\text{iter}}(\text{ProxSkip-LSVRG}).$$

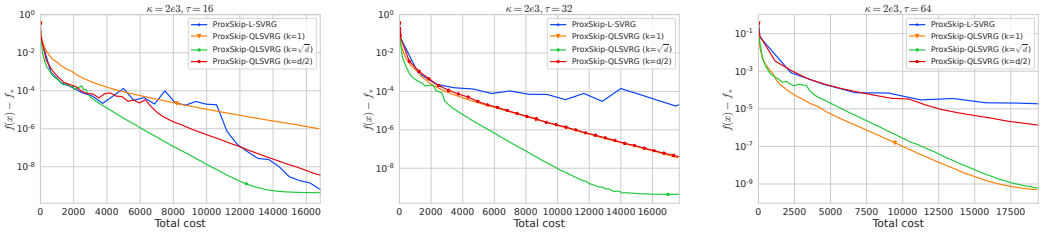
Recall that we assume the communication cost from every worker/hub to the master is equal to 1, and the computation cost per sample is equal to δ. Here we generalize to the multi-level structure. We assume that the communication cost from every client to hub is equal to δ'. We choose the Rand-k sparsification for ProxSkip-HUB; this compressor selects k-entries of the gradient vector, uniformly at random from the full d-dimensional gradient. The total cost of ProxSkip-HUB is

$$\begin{aligned}\text{Cost}(\text{ProxSkip-HUB}) := {} & T_{\text{comm.}}(\text{ProxSkip-HUB}) \\ & + \delta'\left(qm + \frac{k}{d}\left((1-q)\tau + \tau\right)\right) T_{\text{iter}}(\text{ProxSkip-HUB}).\end{aligned} \tag{16}$$

Algorithm 2 ProxSkip-HUB

Input: stepsize $\gamma > 0$, probabilities $p > 0$, $q > 0$, initial iterate $x_0 \in \mathbb{R}^d$, initial shift $y_0 \in \mathbb{R}^d$, initial control variate $h_0 \in \mathbb{R}^d$, number of iterations $T \geq 1$
for $t = 0, 1, \ldots, T-1$ **do**
  broadcast x_t to all clients
  for $i \in S_t$ **do**
    $\hat{\Delta}_t^i = Q\left(\nabla f_i(x_t) - \nabla f_i(y_t)\right)$ ◇ Apply compression operator
  end for
  $\hat{\Delta}_t = \frac{1}{\tau}\sum_{i \in S_t} \hat{\Delta}_t^i$
  $\hat{g} = \hat{\Delta}_t + \nabla f(y_t)$
  $\hat{x}_{t+1} = x_t - \gamma(\hat{g}_t - h_t)$ ◇ Take a gradient-type step adjusted via the control variate h_t
  Flip a coin $\theta_t \in \{0, 1\}$ where $\text{Prob}(\theta_t = 1) = p$ ◇ To decide whether to skip the prox or not
  if $\theta_t = 1$ **then**
    $x_{t+1} = \text{prox}_{\frac{\gamma}{p}\psi}\left(\hat{x}_{t+1} - \frac{\gamma}{p} h_t\right)$ ◇ Apply prox, but only very rarely! (with probability p)
  else
    $x_{t+1} = \hat{x}_{t+1}$ ◇ Skip the prox!
  end if
  $h_{t+1} = h_t + \frac{p}{\gamma}(x_{t+1} - \hat{x}_{t+1})$ ◇ Update the control variate h_t
  $y_{t+1} = \begin{cases} x_t & \text{with probability} \quad q \\ y_t & \text{with probability} \quad 1-q \end{cases}$ ◇ Update the shift y_t
end for

Our experimental results are summarized Figure 11; we use the values $\delta = \delta' = 10^{-2}$. Clearly, and thanks to communication compression, ProxSkip-HUB has benefit in terms of the total cost compared to ProxSkip-LSVRG, and can reach up to three degrees of magnitude!



(a) $\tau = 16$. (b) $\tau = 32$. (c) $\tau = 64$.

Figure 11: Convergence results with different batch sizes and sparsification parameter k on a9a, $\kappa = 2e3$.

B Basic Facts

B.1 Bregman divergence, L-smoothness and μ-strong convexity

The Bregman divergence of a differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is defined by

$$D_f(x, y) := f(x) - f(y) - \langle \nabla f(y), x - y \rangle. \tag{17}$$

It is easy to see that

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle = D_f(x, y) + D_f(y, x), \quad \forall x, y \in \mathbb{R}^d. \tag{18}$$

For an L-smooth and μ-strongly convex function $f : \mathbb{R}^d \to \mathbb{R}$, we have

$$\frac{\mu}{2}\|x - y\|^2 \leq D_f(x, y) \leq \frac{L}{2}\|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d \tag{19}$$

and

$$\frac{1}{2L}\|\nabla f(x) - \nabla f(y)\|^2 \leq D_f(x, y) \leq \frac{1}{2\mu}\|\nabla f(x) - \nabla f(y)\|^2, \quad \forall x, y \in \mathbb{R}^d. \tag{20}$$

B.2 Firm-nonexpansiveness of the proximity operator

Given $\psi : \mathbb{R}^d \to \mathbb{R}$, we define $\psi^*(y) := \sup_{x \in \mathbb{R}^d}\{\langle x, y \rangle - \psi(x)\}$ to be its Fenchel conjugate. The proximity operator of ψ^* satisfies for any $\tau > 0$

$$\text{if } u = \operatorname{prox}_{\tau\psi^*}(y), \quad \text{then} \quad u \in y - \tau\partial\psi^*(u). \tag{21}$$

If Assumption 3 is satisfied, then firm nonexpansiveness of the proximity operator implies [Mishchenko et al., 2022] that

$$\left\|\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \operatorname{prox}_{\frac{\gamma}{p}\psi}(y)\right\|^2 + \left\|\left(x - \operatorname{prox}_{\frac{\gamma}{p}\psi}(x)\right) - \left(y - \operatorname{prox}_{\frac{\gamma}{p}\psi}(y)\right)\right\|^2 \leq \|x - y\|^2, \tag{22}$$

for all $x, y \in \mathbb{R}^d$ and any $\gamma, p > 0$.

B.3 Young's inequality

For any two vectors $a, b \in \mathbb{R}^d$, we have

$$\|a + b\|^2 \leq 2\|a\|^2 + 2\|b\|^2. \tag{23}$$

B.4 Jensen's inequality

For a convex function $h : \mathbb{R}^d \leftarrow \mathbb{R}$ and any vectors $x_1, \ldots, x_n \in \mathbb{R}^d$, we have

$$h\left(\frac{1}{n}\sum_{i=1}^n x_i\right) \leq \frac{1}{n}\sum_{i=1}^n h\left(x_i\right). \tag{24}$$

Applying this to the squared norm, $h(x) = \|x\|^2$, we get

$$\left\|\frac{1}{n}\sum_{i=1}^n y_i\right\|^2 \leq \frac{1}{n}\sum_{i=1}^n \|y_i\|^2. \tag{25}$$

C Analysis of ProxSkip-VR

In this section we provide the proof of Theorem 5.

C.1 Main lemma of ProxSkip

We start from Lemma 1 initially introduced by Mishchenko et al. [2022]; for completeness we provide the whole proof. Let us define two additional sequences:

$$\hat{w}_t = x_t - \gamma \hat{g}_t\left(x_t\right), \qquad \hat{w}_\star = x_\star - \gamma \hat{g}_t\left(x_\star\right). \tag{26}$$

Lemma 1. *If Assumption 3 holds, $\gamma > 0$ and $0 < p \leq 1$, then the iterates of* ProxSkip-VR *satisfy*

$$\mathbb{E}\left[\|x_{t+1} - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_{t+1} - h_\star\|^2\right] \leq \|\hat{w}_t - w_\star\|^2 + \left(1 - p^2\right)\frac{\gamma^2}{p^2}\|h_t - h_\star\|^2. \tag{27}$$

Proof. In order to simplify, let us define two points:

$$x \coloneqq \hat{x}_{t+1} - \frac{\gamma}{p}h_t, \quad y \coloneqq x_\star - \frac{\gamma}{p}h_\star. \tag{28}$$

STEP 1 (Optimality conditions). Using the first-order optimality conditions for $f + \psi$ and using $h_\star \coloneqq \nabla f(x_\star)$, we obtain the following fixed-point identity for $x_\star$:

$$x_\star = \operatorname{prox}_{\frac{\gamma}{p}\psi}\left(x_\star - \frac{\gamma}{p}h_\star\right) = \operatorname{prox}_{\frac{\gamma}{p}\psi}(y). \tag{29}$$

STEP 2 (Recalling the steps of the method). Recall that the vectors x_t and h_t are in Algorithm 1 updated as follows:

$$x_{t+1} = \begin{cases} \operatorname{prox}_{\frac{\gamma}{p}\psi}(x) & \text{with probability} \quad p \\ \hat{x}_{t+1} & \text{with probability} \quad 1-p \end{cases} \tag{30}$$

and

$$h_{t+1} = h_t + \frac{p}{\gamma}\left(x_{t+1} - \hat{x}_{t+1}\right) = \begin{cases} h_t + \frac{p}{\gamma}\left(\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \hat{x}_{t+1}\right) & \text{with probability } p \\ h_t & \text{with probability } 1-p \end{cases}. \tag{31}$$

Let us consider the expected value $V_{t+1} \coloneqq \mathbb{E}\left[\|x_{t+1} - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_{t+1} - h_\star\|^2\right]$:

$$\begin{aligned}
V_{t+1} &\overset{(30)+(31)}{=} p\left(\left\|\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - x_\star\right\|^2 + \frac{\gamma^2}{p^2}\left\|h_t + \frac{p}{\gamma}\left(\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \hat{x}_{t+1}\right) - h_\star\right\|^2\right) \\
&\quad + (1-p)\left(\|\hat{x}_{t+1} - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2\right) \\
&\overset{(29)}{=} p\left(\left\|\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \operatorname{prox}_{\frac{\gamma}{p}\psi}(y)\right\|^2 + \left\|\frac{\gamma}{p}h_t + \operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \hat{x}_{t+1} - \frac{\gamma}{p}h_\star\right\|^2\right) \\
&\quad + (1-p)\left(\|\hat{x}_{t+1} - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2\right) \\
&\overset{(28)+(29)}{=} p\left(\left\|\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - \operatorname{prox}_{\frac{\gamma}{p}\psi}(y)\right\|^2 + \left\|\operatorname{prox}_{\frac{\gamma}{p}\psi}(x) - x + y - \operatorname{prox}_{\frac{\gamma}{p}\psi}(y)\right\|^2\right) \\
&\quad + (1-p)\left(\|\hat{x}_{t+1} - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2\right).
\end{aligned} \tag{32}$$

STEP 4 (Applying firm nonexpansiveness). Applying firm nonexpansiveness of the proximal operator (22), this leads to the inequality

$$\begin{aligned} V_{t+1} &\overset{(32)+(22)}{\leq} p\|x-y\|^2 + (1-p)\left(\|\hat{x}_{t+1}-x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t-h_\star\|^2\right) \\ &\overset{(28)}{=} p\left\|\hat{x}_{t+1} - \frac{\gamma}{p}h_t - \left(x_\star - \frac{\gamma}{p}h_\star\right)\right\|^2 + (1-p)\left(\|\hat{x}_{t+1}-x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t-h_\star\|^2\right). \end{aligned}$$

STEP 5 (Simple algebra). Next, we expand the squared norm and collect the terms, obtaining

$$\begin{aligned} V_{t+1} &\leq p\|\hat{x}_{t+1}-x_\star\|^2 + p\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2 - 2\gamma\langle\hat{x}_{t+1}-x_\star, h_t-h_\star\rangle \\ &\quad +(1-p)\left(\|\hat{x}_{t+1}-x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t-h_\star\|^2\right) \\ &= \|\hat{x}_{t+1}-x_\star\|^2 - 2\gamma\langle\hat{x}_{t+1}-x_\star, h_t-h_\star\rangle + \frac{\gamma^2}{p^2}\|h_t-h_\star\|^2. \end{aligned} \tag{33}$$

Finally, note that by our definition of $\hat{w}_t$, we have the identity $\hat{x}_{t+1} = \hat{w}_t + \gamma h_t$. Therefore, the first two terms above can be rewritten as

$$\begin{aligned} \|\hat{x}_{t+1}-x_\star\|^2 - 2\gamma\langle\hat{x}_{t+1}-x_\star, h_t-h_\star\rangle &= \|\hat{w}_t - w_\star + \gamma(h_t-h_\star)\|^2 \\ &\quad - 2\gamma\langle\hat{w}_t - w_\star + \gamma(h_t-h_\star), h_t-h_\star\rangle \\ &= \|\hat{w}_t-w_\star\|^2 + 2\gamma\langle\hat{w}_t-w_\star, h_t-h_\star\rangle \\ &\quad + \gamma^2\|h_t-h_\star\|^2 - 2\gamma\langle\hat{w}_t-w_\star, h_t-h_\star\rangle \\ &\quad - 2\gamma^2\|h_t-h_\star\|^2 \\ &= \|\hat{w}_t-w_\star\|^2 - \gamma^2\|h_t-h_\star\|^2. \end{aligned} \tag{34}$$

Finally, plugging (34) into (33), we get:

$$V_{t+1} \leq \|\hat{w}_t-w_\star\|^2 + \left(1-p^2\right)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2.$$

□

C.2 Main lemma

This lemma allows us to obtain a useful recursion for variance-reduced stochastic estimators used in our ProxSkip-VR algorithm.

Lemma 2. *Let Assumptions 2 and 4 hold. Then the iterates of* ProxSkip-VR *satisfy*

$$\mathbb{E}\left[\|\hat{w}_t-w_\star\|^2\right] \leq (1-\gamma\mu)\|x_t-x_\star\|^2 - 2\gamma D_f(x_t,x_\star)(1-\gamma A) + \gamma^2 B\sigma_t + \gamma^2 C.$$

Proof. We start from the definitions of the auxiliary sequence $\hat{w}_t$ (see (26)):

$$\begin{aligned} \|\hat{w}_t-w_\star\|^2 &\overset{(26)}{=} \|x_t - \gamma\hat{g}_t - (x_\star - \gamma\nabla f(x_\star))\|^2 \\ &= \|(x_t-x_\star) - \gamma(\hat{g}_t - \nabla f(x_\star))\|^2 \\ &= \|x_t-x_\star\|^2 - 2\gamma\langle x_t-x_\star, \hat{g}_t - \nabla f(x_\star)\rangle + \gamma^2\|\hat{g}_t - \nabla f(x_\star)\|^2. \end{aligned} \tag{35}$$

Taking expectation in (35) and using unbiasedness of $\hat{g}_t$ (see (5) in Assumption 4), we get

$$\mathbb{E}\left[\|\hat{w}_t-w_\star\|^2\right] \overset{(35)+(5)}{=} \|x_t-x_\star\|^2 - 2\gamma\langle x_t-x_\star, \nabla f(x_t) - \nabla f(x_\star)\rangle + \gamma^2\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right]. \tag{36}$$

Let us now consider the inner product in (36). Using (19) and (18), we obtain

$$\mathbb{E}\left[\|\hat{w}_t-w_\star\|^2\right] \leq (1-\gamma\mu)\|x_t-x_\star\|^2 - 2\gamma D_f(x_t,x_\star) + \gamma^2\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right]. \tag{37}$$

To bound the last term in (37), we can apply Assumption 4:

$$\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right] \leq 2AD_f(x_t,x_\star) + B\sigma_t + C. \tag{38}$$

Plugging (38) into (37) gives us

$$\begin{aligned} \mathbb{E}\left[\|\hat{w}_t-w_\star\|^2\right] &\leq (1-\gamma\mu)\|x_t-x_\star\|^2 - 2\gamma D_f(x_t,x_\star) + \gamma^2\left(2AD_f(x_t,x_\star) + B\sigma_t + C\right) \\ &\leq (1-\gamma\mu)\|x_t-x_\star\|^2 - 2\gamma D_f(x_t,x_\star)(1-\gamma A) + \gamma^2 B\sigma_t + \gamma^2 C, \end{aligned} \tag{39}$$

which is what we set out to prove. □

C.3 Proof of Theorem 5

Proof. Using definition of the Lyapunov function Ψ_t, and the tower property of conditional expectation, we obtain

$$\begin{aligned}
\mathbb{E}\left[\Psi_{t+1}\right] &= \mathbb{E}\left[\|x_{t+1}-x_\star\|^2+\frac{\gamma^2}{p^2}\|h_{t+1}-h_\star\|^2+W\gamma^2\sigma_{t+1}\right] \\
&\overset{(27)}{\leq} \mathbb{E}\left[\|\hat{w}_t-w_\star\|^2\right]+(1-p^2)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2 \\
&\quad +W\gamma^2\left(2\tilde{A}D_f(x_t,x_\star)+\tilde{B}\sigma_t+\tilde{C}\right) \\
&\overset{(39)}{\leq} (1-\gamma\mu)\|x_t-x_\star\|^2-2\gamma D_f(x_t,x_\star)\left(1-\gamma A\right)+\gamma^2B\sigma_t+\gamma^2C \\
&\quad +(1-p^2)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2+W\gamma^2\left(2\tilde{A}D_f(x_t,x_\star)+\tilde{B}\sigma_t+\tilde{C}\right) \\
&\leq (1-\gamma\mu)\|x_t-x_\star\|^2-2\gamma D_f(x_t,x_\star)\left(1-\gamma(A+W\tilde{A})\right) \\
&\quad +\gamma^2W\sigma_t\left(\frac{B+W\tilde{B}}{W}\right)+\gamma^2(C+W\tilde{C})+(1-p^2)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2. \quad (40)
\end{aligned}$$

Using the stepsize bound $\gamma \leq \frac{1}{A+W\tilde{A}}$, this leads to

$$\mathbb{E}\left[\Psi_{t+1}\right] \leq (1-\gamma\mu)\|x_t-x_\star\|^2+\gamma^2W\sigma_t\left(\frac{B+W\tilde{B}}{W}\right)+\gamma^2(C+W\tilde{C})+(1-p^2)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2.$$

Let us denote $\beta \coloneqq \frac{B+W\tilde{B}}{W}$. In order to obtain a contraction, we need to have $\beta < 1$, which is satisfied when $W > \frac{B}{1-\tilde{B}}$, and we get

$$\begin{aligned}
\mathbb{E}\left[\Psi_{t+1}\right] &\leq (1-\gamma\mu)\|x_t-x_\star\|^2+\gamma^2W\sigma_t\beta+\gamma^2(C+W\tilde{C})+(1-p^2)\frac{\gamma^2}{p^2}\|h_t-h_\star\|^2 \\
&\leq \max\left(1-p^2,\beta,1-\gamma\mu\right)\Psi_t+\gamma^2(C+W\tilde{C}). \quad (41)
\end{aligned}$$

Finally, using the tower property of expectation and unrolling recursion (41), we get

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T,\beta^T,(1-p^2)^T\right\}\Psi_0+\frac{\left(C+W\tilde{C}\right)\gamma^2}{\min\left\{\gamma\mu,p^2,1-\beta\right\}}.$$

□

D Examples of Methods Without Variance Reduction

D.1 Proof of Theorem 6 (GD estimator)

Proof. Let us show that GD estimator ($\hat{g}_t = \nabla f(x_t)$) satisfies Assumption 4

$$\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right] = \|\nabla f(x_t) - \nabla f(x_\star)\|^2 \overset{(19)}{\leq} 2LD_f(x_t, x_\star).$$

This means that Assumption 4 is satisfied with the following constant:

$$A = L, \quad B = 0, \quad C = 0, \quad \tilde{A} = 0, \quad \tilde{B} = 0, \quad \tilde{C} = 0, \quad \sigma_t \equiv 0.$$

Applying Theorem 5 leads to final recursion:

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T\right\}\Psi_0, \tag{42}$$

By inspecting (42) it is easy to see that

$$T \geq \max\left\{\frac{1}{\gamma\mu}, \frac{1}{p^2}\right\}\log\frac{1}{\varepsilon} \qquad \Longrightarrow \qquad \mathbb{E}\left[\Psi_T\right] \leq \varepsilon\Psi_0. \tag{43}$$

Then the communication complexity is equal to

$$pT \geq \max\left\{\frac{p}{\gamma\mu}, \frac{1}{p}\right\}\log\frac{1}{\varepsilon}. \tag{44}$$

Setting $\gamma = \frac{1}{L}$ and solving $\frac{pL}{\mu} = \frac{1}{p}$ gives the optimal probability

$$p = \sqrt{\frac{\mu}{L}} = \frac{1}{\sqrt{\kappa}} \tag{45}$$

Finally, the iteration complexity and communication complexity have the following form:

$$T \geq \max\left\{\frac{1}{\gamma\mu}, \frac{1}{p^2}\right\}\log\frac{1}{\varepsilon} = \kappa\log\frac{1}{\varepsilon}, \tag{46}$$

$$pT \geq \max\left\{\frac{p}{\gamma\mu}, \frac{1}{p}\right\}\log\frac{1}{\varepsilon} = \sqrt{\kappa}\log\frac{1}{\varepsilon}. \tag{47}$$

□

This recovers the result obtained by Mishchenko et al. [2022].

D.2 Proof of Theorem 8 (SGD estimator)

Proof. Let us show that the SGD estimator $\hat{g}_t = g(x_t, \xi_t)$ satisfying Assumption 7 also satisfies Assumption 4. Using Young's inequality we get

$$\begin{aligned}
\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right] &= \mathbb{E}\left[\|g(x_t,\xi_t) - \nabla f(x_\star)\|^2\right] \\
&= \mathbb{E}\left[\|g(x_t,\xi_t) - g(x_\star,\xi_t) + g(x_\star,\xi_t) - \nabla f(x_\star)\|^2\right] \\
&\overset{(23)}{\leq} 2\mathbb{E}\left[\|g(x_t,\xi_t) - g(x_\star,\xi_t)\|^2\right] + 2\mathbb{E}\left[g(x_\star,\xi_t) - \nabla f(x_\star)\|^2\right] \\
&\overset{(7)}{\leq} 4A''D_f(x_t,x_\star) + 2\mathrm{Var}(g(x_\star,\xi)).
\end{aligned} \tag{48}$$

This means that Assumption 4 is satisfied with the following constants:

$$A = 2A'', \quad B = 0, \quad C = 2\mathrm{Var}(g(x_\star,\xi)), \quad \tilde{A} = 0, \quad \tilde{B} = 0, \quad \tilde{C} = 0, \quad \sigma_t \equiv 0.$$

Applying Theorem 5 leads to the final bound:

$$\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T\right\}\Psi_0 + \gamma^2\frac{2\mathrm{Var}(g(x_\star,\xi))}{\min\left\{\gamma\mu, p^2\right\}}. \tag{49}$$

In order to minimize the number of prox evaluations, whatever the choice of γ will be, we choose the smallest probability p which does not lead to any degradation of the rate $\min\{\gamma\mu, p^2\}$. That is, we choose $p = \sqrt{\gamma\mu}$. The first term on the right-hand side of (49) can be bounded as follows:

$$T \geq \frac{1}{\gamma\mu} \log\left(\frac{2\Psi_0}{\varepsilon}\right) \quad \Longrightarrow \quad (1-\gamma\mu)^T \Psi_0 \leq \frac{\varepsilon}{2}.$$

The second term on the right-hand side of (49) can be bounded as follows:

$$\gamma \leq \frac{\varepsilon\mu}{2C} \quad \Longrightarrow \quad \frac{\gamma C}{\mu} \leq \frac{\varepsilon}{2}.$$

We choose the largest stepsize consistent with bounds $\gamma \leq \frac{\varepsilon\mu}{2C}$ and $\gamma \leq \frac{1}{A}$:

$$\gamma = \min\left\{\frac{1}{A}, \frac{\varepsilon\mu}{2C}\right\}.$$

Using this stepsizem we get the following iteration and (expected) communication complexities:

$$T \geq \max\left\{\frac{A}{\mu}, \frac{2C}{\varepsilon\mu^2}\right\} \log\left(\frac{2\Psi_0}{\varepsilon}\right), \qquad pT \geq \max\left\{\sqrt{\frac{A}{\mu}}, \sqrt{\frac{2C}{\varepsilon\mu^2}}\right\} \log\left(\frac{2\Psi_0}{\varepsilon}\right).$$

This recovers the result obtained by Mishchenko et al. [2022]. □

E Analysis of ProxSkip-HUB

In this section we provide analysis of the new algorithm ProxSkip-HUB, which works for the new FL formulation described in Section 4. The pseudocode is presented in Algorithm 2.

E.1 Lemma for minibatch sampling

Fix a minibatch size $\tau \in \{1, 2, \ldots, n\}$ and let $\mathcal{S}_t$ be a random subset of $\{1, 2, \ldots, n\}$ of size τ, chosen uniformly at random. Define the gradient estimator via

$$g(x) := \frac{1}{\tau} \sum_{j \in \mathcal{S}_t} \nabla \widetilde{\phi}_j(x) \tag{50}$$

Lemma 3. *The gradient estimator $g(x)$ defined in (50) is unbiased. If we further assume that $n \geq 2$, $\widetilde{\phi}_j$ is convex and L_j-smooth for all j, and f is L-smooth, then*

$$\mathbb{E}\left[\|g(x_t) - g(x_\star)\|^2\right] \leq 2L(\tau) D_f(x_t, x_\star),$$

where

$$L(\tau) := \frac{n-\tau}{\tau(n-1)} \max_j L_j + \frac{n(\tau-1)}{\tau(n-1)} L.$$

Proof. Let χ_j be the random variable defined by

$$\chi_j = \begin{cases} 1 & j \in S \\ 0 & j \notin S \end{cases}.$$

It is easy to show that

$$\mathbb{E}\left[\chi_j\right] = \text{Prob}(j \in S) = \frac{\tau}{n}. \tag{51}$$

Unbiasedness of $g(x)$ now follows via direct computation:

$$\begin{aligned} \mathbb{E}\left[g(x)\right] &\overset{(50)}{=} \mathbb{E}\left[\frac{1}{\tau} \sum_{j \in S} \nabla \widetilde{\phi}_j(x)\right] = \mathbb{E}\left[\frac{1}{\tau} \sum_{j=1}^n \chi_i \nabla \widetilde{\phi}_j(x)\right] = \frac{1}{\tau} \sum_{j=1}^n \mathbb{E}\left[\chi_i\right] \nabla \widetilde{\phi}_j(x) \\ &= \frac{1}{\tau} \sum_{j=1}^n \text{Prob}(j \in S) \nabla \widetilde{\phi}_j(x) \overset{(51)}{=} \frac{1}{\tau} \sum_{j=1}^n \frac{\tau}{n} \nabla \widetilde{\phi}_j(x) = \nabla f(x). \end{aligned}$$

Let us define

$$a_j := \nabla \widetilde{\phi}_j(x) - \nabla \widetilde{\phi}_j(x_\star). \tag{52}$$

Let $\chi_{j,k}$ be the random variable defined by

$$\chi_{j,k} = \begin{cases} 1 & j \in \mathcal{S}_t \text{ and } k \in \mathcal{S}_t \\ 0 & \text{otherwise} \end{cases}.$$

Note that

$$\chi_{j,k} = \chi_j \chi_k. \tag{53}$$

Further, it is easy to show that

$$\mathbb{E}\left[\chi_{j,k}\right] = \text{Prob}(j \in \mathcal{S}_t, k \in \mathcal{S}_t) = \frac{\tau(\tau-1)}{n(n-1)}. \tag{54}$$

Let us consider

$$
\begin{aligned}
\mathbb{E}\left[\|g(x_t)-g(x_\star)\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]\\
&= \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\left(\nabla\widetilde{\phi}_j(x)-\nabla\widetilde{\phi}_j(x_\star)\right)\right\|^2\right]\\
&\overset{(52)}{=} \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}a_j\right\|^2\right]\\
&= \frac{1}{\tau^2}\mathbb{E}\left[\left\|\sum_{j=1}^n\chi_j a_j\right\|^2\right]\\
&= \frac{1}{\tau^2}\mathbb{E}\left[\sum_{j=1}^n\|\chi_j a_j\|^2+\sum_{k\neq j}\langle\chi_j a_j,\chi_k a_k\rangle\right]\\
&\overset{(53)}{=} \frac{1}{\tau^2}\mathbb{E}\left[\sum_{j=1}^n\|\chi_j a_j\|^2+\sum_{k\neq j}\chi_{j,k}\langle a_j,a_k\rangle\right].
\end{aligned}
\tag{55}
$$

Using the formulas (51) and (54) we can continue:

$$
\begin{aligned}
\mathbb{E}\left[\|g(x_t)-g(x_\star)\|^2\right] &= \frac{1}{\tau^2}\left(\frac{\tau}{n}\sum_{j=1}^n\|a_j\|^2+\frac{\tau(\tau-1)}{n(n-1)}\sum_{j\neq k}\langle a_j,a_k\rangle\right)\\
&= \frac{1}{\tau n}\sum_{j=1}^n\|a_j\|^2+\frac{\tau-1}{\tau n(n-1)}\sum_{j\neq k}\langle a_j,a_k\rangle\\
&= \frac{1}{\tau n}\sum_{j=1}^n\|a_j\|^2+\frac{\tau-1}{\tau n(n-1)}\left(\left\|\sum_{j=1}^n a_j\right\|^2-\sum_{j=1}^n\|a_j\|^2\right)\\
&= \frac{n-\tau}{\tau(n-1)}\frac{1}{n}\sum_{j=1}^n\|a_j\|^2+\frac{n(\tau-1)}{\tau(n-1)}\left\|\frac{1}{n}\sum_{j=1}^n a_j\right\|^2.
\end{aligned}
\tag{56}
$$

Since $\widetilde{\phi}_j$ is convex and L_j-smooth, we know that

$$
\|a_j\|^2=\|\nabla f_j(x_t)-\nabla f_j(x_\star)\|^2\leq 2L_jD_{\widetilde{\phi}_j}(x_t,x_\star). \tag{57}
$$

Since f is convex and L-smooth, we know that

$$
\left\|\frac{1}{n}\sum_{i=1}^n a_i\right\|^2=\|\nabla f(x_t)-\nabla f(x_\star)\|^2\leq 2LD_f(x_t,x_\star). \tag{58}
$$

Let us apply the bound $L_j \leq \max_j L_j$ and use the identity $D_f(x_t, x_\star) = \frac{1}{n}\sum_{j=1}^n D_{\widetilde{\phi}_j}(x_t, x_\star)$:

$$
\begin{aligned}
\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x) - \frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right] &\leq \frac{n-\tau}{\tau(n-1)}\frac{1}{n}\sum_{j=1}^n 2L_j D_{\widetilde{\phi}_j}(x_t, x_\star) \\
&\quad + \frac{n(\tau-1)}{\tau(n-1)}2LD_f(x_t, x_\star) \\
&\leq 2\frac{n-\tau}{\tau(n-1)}\max_j L_j \frac{1}{n}\sum_{j=1}^n D_{f_j}(x_t, x_\star) \\
&\quad + 2\frac{n(\tau-1)}{\tau(n-1)}LD_f(x, y) \\
&= 2\frac{n-\tau}{\tau(n-1)}\max_j L_j D_f(x, y) \\
&\quad + 2\frac{n(\tau-1)}{\tau(n-1)}LD_f(x_t, x_\star) \\
&= 2\left(\frac{n-\tau}{\tau(n-1)}\max_j L_j + \frac{n(\tau-1)}{\tau(n-1)}L\right)D_f(x_t, x_\star).
\end{aligned}
$$

□

E.2 Proof of Theorem 9

As in previous analysis we need to show that Assumption 4 is satisfied for the ProxSkip-HUB method.

Proof. Let us consider the first inequality in Assumption 4 and show that it holds for the new gradient estimator $\hat{g}_t = \frac{1}{\tau}\sum_{j\in\mathcal{S}_t} Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) + \nabla f(y)$:

$$
\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right] = \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t} Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) + \nabla f(y_t) - \nabla f(x_\star)\right\|^2\right]. \tag{59}
$$

Let $\Delta_t = \frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))$ and $\hat{\Delta}_t = \frac{1}{\tau}\sum_{j\in\mathcal{S}_t} Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))$. Using smart zero $0 = \hat{\Delta}_t - \Delta_t$ we have

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{g}_t - \nabla f(x_\star)\|^2\right] &= \mathbb{E}\left[\left\|\hat{\Delta}_t - \Delta_t + \Delta_t + \nabla f(y) - \nabla f(x_\star)\right\|^2\right] \\
&\overset{(23)}{\leq} 2\mathbb{E}\left[\|\hat{\Delta}_t - \Delta_t\|^2\right] + 2\mathbb{E}\left[\|\Delta_t + \nabla f(y_t) - \nabla f(x_\star)\|^2\right] \\
&\leq 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t} Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) - \frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))\right\|^2\right] \\
&\quad + 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) + \nabla f(y) - \nabla f(x_\star)\right\|^2\right].
\end{aligned} \tag{60}
$$

Let us consider the first term (60), let us define $\theta_t^i = Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) - (\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))$:

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{\Delta}_t - \Delta_t\|^2\right] &= \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t} Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) - (\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t} \theta_t^i\right\|^2\right] \\
&= \mathbb{E}\left[\frac{1}{\tau^2}\left(\sum_{j\in\mathcal{S}_t}\|\theta_t^i\|^2 + \sum_{i\neq j}\left\langle\theta_t^i, \theta_t^j\right\rangle\right)\right] \\
&= \frac{1}{\tau^2}\left(\sum_{j\in\mathcal{S}_t}\mathbb{E}\left[\|\theta_t^i\|^2\right] + \sum_{i\neq j}\mathbb{E}\left[\left\langle\theta_t^i, \theta_t^j\right\rangle\right]\right).
\end{aligned}
$$

Using independence and unbiasedness of compressors we have

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{\Delta}_t - \Delta_t\|^2\right] &= \frac{1}{\tau^2}\left(\sum_{j\in\mathcal{S}_t}\mathbb{E}\left[\|\theta_t^i\|^2\right] + \sum_{i\neq j}\mathbb{E}\left[\left\langle\theta_t^i, \theta_t^j\right\rangle\right]\right) \\
&= \frac{1}{\tau^2}\left(\sum_{j\in\mathcal{S}_t}\mathbb{E}\left[\|\theta_t^i\|^2\right] + \sum_{i\neq j}\left\langle\mathbb{E}\left[\theta_t^i\right], \mathbb{E}\left[\theta_t^j\right]\right\rangle\right) \\
&= \frac{1}{\tau^2}\sum_{j\in\mathcal{S}_t}\mathbb{E}\left[\|\theta_t^i\|^2\right] \\
&= \frac{1}{\tau^2}\sum_{j\in\mathcal{S}_t}\mathbb{E}\left[\|Q(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)) - (\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t))\|^2\right] \\
&\overset{(12)}{\leq} \frac{\omega}{\tau}\mathbb{E}\left[\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\|\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(y_t)\|^2\right]. \qquad (61)
\end{aligned}
$$

Using Young's inequality and expectation of client sampling we get

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{\Delta}_t - \Delta_t\|^2\right] &\overset{(23)}{\leq} \frac{2\omega}{\tau}\mathbb{E}\left[\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\|\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(x_\star)\|^2\right] \\
&\quad + \frac{2\omega}{\tau}\mathbb{E}\left[\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\|\nabla\widetilde{\phi}_j(y_t) - \nabla\widetilde{\phi}_j(x_\star)\|^2\right] \\
&\overset{(23)}{\leq} \frac{2\omega}{\tau}\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(x_\star)\|^2 \\
&\quad + \frac{2\omega}{\tau}\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(y_t) - \nabla\widetilde{\phi}_j(x_\star)\|^2 \\
&\overset{(20)}{\leq} \frac{4\omega}{\tau}L_{\max}D_f(x_t, x_\star) + \frac{2\omega}{\tau}\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(y_t) - \nabla\widetilde{\phi}_j(x_\star)\|^2. \qquad (62)
\end{aligned}
$$

Let us consider the second term in (60):

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t))+\nabla f(y)-\nabla f(x_\star)\right\|^2\right]\\
&=\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t)+\nabla\widetilde{\phi}_j(x_\star)-\nabla\widetilde{\phi}_j(x_\star))+\nabla f(y)-\nabla f(x_\star)\right\|^2\right]\\
&\overset{(23)}{\leq} 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]\\
&\quad+2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(y_t)-\left(\nabla f(x_\star)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(y_t)\right)\right\|^2\right]\\
&\leq 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]+2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(y_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right].
\end{aligned}
\tag{63}
$$

Using Lemma 3 and Jensen's inequality (25) we have

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t))+\nabla f(y)-\nabla f(x_\star)\right\|^2\right]\\
&\leq 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]+2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(y_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]\\
&\leq 4L(\tau)D_f(x_t,x_\star)+\frac{2}{n}\sum_{j=1}^n\left\|\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star)\right\|^2.
\end{aligned}
\tag{64}
$$

Combining two parts (62), (64) and plugging into (60) we get

$$
\begin{aligned}
\mathbb{E}\left[\|\hat{g}_t-\nabla f(x_\star)\|^2\right]&\leq 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}Q(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t))-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t))\right\|^2\right]\\
&\quad+2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t))+\nabla f(y)-\nabla f(x_\star)\right\|^2\right]\\
&\leq 8L(\tau)D_f(x_t,x_\star)+\frac{4}{n}\sum_{j=1}^n\left\|\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star)\right\|^2\\
&\quad+\frac{8\omega}{\tau}L_{\max}D_f(x_t,x_\star)+\frac{4\omega}{\tau}\frac{1}{n}\sum_{j=1}^n\|\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star)\|^2\\
&\leq 2\cdot 4\left(L(\tau)+\frac{\omega}{\tau}L_{\max}\right)D_f(x_t,x_\star)+4\left(1+\frac{\omega}{\tau}\right)\sigma_t,
\end{aligned}
\tag{65}
$$

where $\sigma_t=\frac{1}{n}\sum_{j=1}^n\|\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star)\|^2$. Let us consider update of control variable y_t:

$$
y_{t+1}=\begin{cases} x_t & \text{with probability} \quad q\\ y_t & \text{with probability} \quad 1-q\end{cases}. \tag{66}
$$

Let us show that second inequality in Assumption 4:

$$\begin{aligned}\mathbb{E}\left[\sigma_{t+1}\right] &= \mathbb{E}\left[\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(y_{t+1})-\nabla\widetilde{\phi}_j(x_\star)\|^2\right]\\ &= (1-q)\,\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star)\|^2 + q\frac{1}{n}\sum_{j=1}^{n}\|\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(x_\star)\|^2\\ &= (1-q)\,\sigma_t + 2qL_{\max}D_f(x_t,x_\star). \end{aligned} \tag{67}$$

Using (65) and (72) bounds we can confirm that Assumption 4 is satisfied with the following constants:

$$A = 4\left(L(\tau)+\frac{\omega}{\tau}L_{\max}\right),\quad B = 4\left(1+\frac{\omega}{\tau}\right),\quad C=0,\quad \tilde{A}=qL_{\max},\quad \tilde{B}=1-q,\quad \tilde{C}=0.$$

Applying Theorem 5 leads to the final result

$$\mathbb{E}\left[\Psi_T\right] \le \max\left\{(1-\gamma\mu)^T,(1-p^2)^T,(1-q/2)^T\right\}\Psi_0, \tag{68}$$

where the Lyapunov function is defined by

$$\Psi_t := \|x_t-x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t-h_\star\|^2 + \gamma^2\frac{8}{q}\left(1+\frac{\omega}{\tau}\right)\sigma_t.$$

Let us set $\gamma = \frac{1}{A+W\tilde{A}}$, $p=\sqrt{\gamma\mu}$ and $q=2\gamma\mu$. Using the same proof as for ProxSkip in Section D.1 and $L(\tau)\le L_{\max}$ we get communication and iteration complexities:

$$T_{\text{comm.}} = \mathcal{O}\left(\sqrt{\frac{L_{\max}}{\mu}\left(1+\frac{\omega}{\tau}\right)}\log\frac{1}{\varepsilon}\right),\qquad T_{\text{iter.}} = \mathcal{O}\left(\frac{L_{\max}}{\mu}\left(1+\frac{\omega}{\tau}\right)\log\frac{1}{\varepsilon}\right).$$

□

If we use full participation $\tau=n$ and $q=\frac{1}{\omega+1}$ and $r(x)\equiv 0$ then we get the same rate as for DIANA [Mishchenko et al., 2019, Horváth et al., 2019b] and RAND-DIANA [Shulgin and Richtárik, 2021].

F Analysis of ProxSkip-LSVRG

The analysis of ProxSkip-LSVRG is almost the same to the analysis of ProxSkip-HUB, with one exception. We use a different sigma component:

$$\sigma_t = \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}(\nabla\widetilde{\phi}_j(y_t)-\nabla\widetilde{\phi}_j(x_\star))\right\|^2\right]. \tag{69}$$

Let us consider $\mathbb{E}\left[\|\hat{g}_t-\nabla f(x_\star)\|^2\right]$:

$$\mathbb{E}\left[\|\hat{g}_t-\nabla f(x_\star)\|^2\right] = \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in S_t}\left(\nabla\widetilde{\phi}_j(x_t)-\nabla\widetilde{\phi}_j(y_t)\right)+\nabla f(y)-\nabla f(x_\star)\right\|^2\right] \tag{70}$$

Using (63) we have

$$\begin{aligned}\mathbb{E}\left[\|\hat{g}_t-\nabla f(x_\star)\|^2\right] &\le 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]\\ &\quad + 2\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(y_t)-\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\nabla\widetilde{\phi}_j(x_\star)\right\|^2\right]\\ &\le 4L(\tau)D_f(x_t,x_\star)+2\sigma_t. \end{aligned} \tag{71}$$

Let us show that second inequality in Assumption 4, using Lemma 3 we get

$$
\begin{aligned}
\mathbb{E}\left[\sigma_{t+1}\right] &= \mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\left(\nabla\widetilde{\phi}_j(y_{t+1}) - \nabla\widetilde{\phi}_j(x_\star)\right)\right\|^2\right] \\
&= (1-q)\,\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\left(\nabla\widetilde{\phi}_j(y_t) - \nabla\widetilde{\phi}_j(x_\star)\right)\right\|^2\right] + q\mathbb{E}\left[\left\|\frac{1}{\tau}\sum_{j\in\mathcal{S}_t}\left(\nabla\widetilde{\phi}_j(x_t) - \nabla\widetilde{\phi}_j(x_\star)\right)\right\|^2\right] \\
&= (1-q)\,\sigma_t + 2qL(\tau)D_f(x_t, x_\star). \qquad (72)
\end{aligned}
$$

We showed that Assumption 4 is satisfied with following constants:

$$
A = 2L(\tau), \quad B = 2, \quad C = 0, \quad \tilde{A} = qL(\tau), \quad \tilde{B} = 1-q, \quad \tilde{C} = 0. \qquad (73)
$$

Applying Theorem 5 with $\gamma = \frac{1}{6L(\tau)}$ we get final bound:

$$
\mathbb{E}\left[\Psi_T\right] \leq \max\left\{(1-\gamma\mu)^T, (1-p^2)^T, \left(1-\frac{q}{2}\right)^T\right\}\Psi_0,
$$

where the Lyapunov function is defined as

$$
\Psi_t \coloneqq \|x_t - x_\star\|^2 + \frac{\gamma^2}{p^2}\|h_t - h_\star\|^2 + \gamma^2\frac{4}{q}\sigma_t.
$$

Using the same argument as for ProxSkip and setting $\frac{q}{2} = \gamma\mu$, we get

$$
T_{\text{comms}} = \mathcal{O}\left(\sqrt{\frac{L(\tau)}{\mu}}\log\frac{1}{\varepsilon}\right), \qquad T_{\text{iters}} = \mathcal{O}\left(\frac{L(\tau)}{\mu}\log\frac{1}{\varepsilon}\right). \qquad (74)
$$

If $r(x) \equiv 0$, this recovers results of Kovalev et al. [2020a].