

Fastest Rates for Stochastic Mirror Descent Methods*

Filip Hanzely[†]

Peter Richtárik[‡]

March 20, 2018

Abstract

Relative smoothness - a notion introduced in [6] and recently rediscovered in [3, 18] - generalizes the standard notion of smoothness typically used in the analysis of gradient type methods. In this work we are taking ideas from well studied field of stochastic convex optimization and using them in order to obtain faster algorithms for minimizing relatively smooth functions. We propose and analyze two new algorithms: Relative Randomized Coordinate Descent (relRCD) and Relative Stochastic Gradient Descent (relSGD), both generalizing famous algorithms in the standard smooth setting. The methods we propose can be in fact seen as a particular instances of stochastic mirror descent algorithms. One of them, relRCD corresponds to the first stochastic variant of mirror descent algorithm with linear convergence rate.

1 Introduction

During the last decade or so, *first order methods* have become the main algorithmic toolbox for practitioners solving optimization problems of large sizes, especially in application domains where low to medium accuracy is sufficient. These methods are now the state of the art for many problems arising in areas such as machine learning, statistics, signal processing, computer vision, inverse problems and data science. Arguably, algorithms for *smooth convex optimization* form the backbone of this new development, and the basis for subsequent extensions beyond convexity and smoothness.

In this paper we consider the optimization problem

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & x \in Q, \end{aligned} \tag{1}$$

where $Q \subseteq \mathbb{R}^n$ is a closed convex set, and f is a convex and differentiable¹ (objective/loss) function.

Our work is motivated by the need to solve problems of the form (1) in the “big data” regime, that is, in situations when either the dimensionality of the problem, n , is very large, or when f is of a finite sum structure,

$$f(x) = \frac{1}{m} \sum_{i=1}^m f_i(x), \tag{2}$$

*All theoretical results of this paper were obtained by June 2017.

[†]King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia

[‡]King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia — University of Edinburgh, Edinburgh, United Kingdom — Moscow Institute of Physics and Technology (MIPT), Dolgoprudny, Russia

¹We assume that f is differentiable on some open set containing Q .

with the number of components, m , being very large. In particular, we are interested in designing efficient *randomized* first order methods for (1) without the need to assume for f to have Lipschitz gradients, thus extending the reach of modern randomized gradient-type methods to new territories.

1.1 Lipschitz continuity of the gradients

It is remarkable that virtually the entire development of first order methods for smooth convex optimization hinges on what turns out to be a very restrictive regularity assumption on the behaviour of the gradients of f , thus preventing their applicability to domains where this assumption does not hold, or is unreasonable due to practical considerations. In particular, it is universally assumed for the objective function f to have Lipschitz continuous gradients [21, 26, 24]. Recall that f is said to be L -smooth on Q (equivalently, we say that *the gradient of f is L -Lipschitz on Q*), if

$$f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2} \|x - y\|^2, \quad \text{for all } x, y \in Q, \quad (3)$$

where $\langle \cdot, \cdot \rangle$ is an inner product and $\|x\| = \langle x, x \rangle^{1/2}$ is the induced norm².

The archetypal first order method for solving (1), *projected gradient descent* (GD), is designed to take advantage of the approximation (3). Given $x_t \in Q$, the next iterate x_{t+1} of GD is obtained by minimizing the upper *quadratic* bound on f provided by (3) for $y = x_t$:

$$x_{t+1} = \arg \min_{x \in Q} \langle \nabla f(x_t), x - x_t \rangle + \frac{L}{2} \|x - x_t\|^2$$

That is, in the design of GD, one employs a majorize-minimize approach [13].

However, there are many differentiable convex functions which are not L -smooth for any finite L . For instance, consider the function $f(x) = x^4$ on \mathbb{R} . If we still wish to apply a gradient type method to minimize such a function, L -smoothness can sometimes be forced upon f by introducing appropriate constrains. This is sufficient in principle as the theory for constrained first order methods only requires the gradients to be L -Lipschitz on the domain of interest. However, such a restriction often leads to a very large constant L in practice, which leads to a prohibitive slow-down of the methods, unless line search strategies are used. Indeed, the performance of first order type methods deteriorates as L grows, typically at a linear or quadratic rate. Moreover, even if the objective is naturally L -smooth, the constant L is often very large, reflecting poor conditioning of the problem. In all these cases, direct application of first-order machinery is either impossible or prohibitively inefficient, which leaves these problems beyond the reach of some of the most efficient algorithms designed for large problems in the last decade.

1.2 Relative smoothness: beyond Lipschitz continuity

Relative smoothness was first introduced in [6] and later rediscovered independently [3] and [18]

This notion enables to design and analyze a generalized version of GD which we refer to in this paper as *relative gradient descent* (relGD). We shall now briefly outline their approach.

Let $h : Q \rightarrow \mathbb{R}$ be a strictly convex and differentiable function. The Bregman distance (divergence) of h is the function

$$D_h(x, y) \stackrel{\text{def}}{=} h(x) - h(y) - \langle \nabla h(y), x - y \rangle. \quad (4)$$

²An equivalent characterization of L -smoothness is to require the inequality $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ to hold for all $x, y \in Q$.

Clearly, $D_h(x, y) \geq 0$ and $D_h(x, y) = 0$ if and only if $x = y$. However, D_h is not necessarily symmetric.

In analogy with (3), Bauschke et al [3] and Lu et al [18] say that f is L -smooth relative to h on Q if

$$f(x) \leq f(y) + \langle \nabla f(y), x - y \rangle + LD_h(x, y), \quad \text{for all } x, y \in Q. \quad (5)$$

In analogy with the design of gradient descent, relative gradient descent minimizes the upper bound on f given by (5) for $y = x_t$:

$$x_{t+1} = \arg \min_{x \in Q} \langle \nabla f(x_t), x - x_t \rangle + LD(x, x_t) \quad (6)$$

Note that if $h(x) = \frac{1}{2}\|x\|^2$, then $D_h(x, y) = \frac{1}{2}\|x - y\|^2$, and L -smoothness relative to h defined in (5) coincides with standard L -smoothness defined in (3). Likewise, relative gradient descent coincides with gradient descent.

1.3 Introducing randomness

For problems of truly huge sizes (if, as alluded to earlier, either m or n are very large), randomized first order methods, such as variants of stochastic gradient descent [34, 20, 36, 35, 37, 14] (in case of large m) and randomized coordinate descent (in case of large n) [22, 32, 33, 37, 27], have become the methods of choice, both in theory and in practice.

While a single iteration of a randomized method typically leads to small improvement relative to the improvement obtained by its deterministic counterpart, stochastic iterations are in general much faster: for problems of suitable structure, each iteration is typically n (for randomized coordinate descent type methods) or m (for stochastic gradient descent type methods) times faster than one iteration of gradient descent. The trade-off is in favour of stochastic methods: the savings obtained by performing faster iterations outweigh the loss incurred by settling with smaller per-iteration improvements.

1.4 Contributions

In this paper we develop the first stochastic algorithms for minimizing relatively smooth functions. In so doing, we push the boundary of big data optimization beyond the realm of L -smoothness.

All methods developed in this work are of the form

$$x_{t+1} = \operatorname{argmin}_{x \in Q_t} \{ \langle g_t, x \rangle + L_t D_h(x, x_t) \} \quad (7)$$

for suitable set $Q_t \subset \mathbb{R}^n$, vector $g_t \in \mathbb{R}^n$ and a sequence of stepsizes $\{L_t\}$. Note that by choosing $g_t = \nabla f(x_t)$, $L_t = L$ and $Q_t = Q$, we obtain method (6), i.e., relative gradient descent [3, 18].

We prove convergence of different success measures, including expected suboptimality in the objective, Bregman distance to the optimum, and Bregman distance between iterates. Below we briefly outline some of the results obtained.

Our algorithms belong to two categories:

Relative Randomized Coordinate Descent (relRCD). This arises as a special of the generic method (7) if we choose $g_t = \nabla f(x_t)$, pick suitable stepsizes L_t , and let Q_t correspond to a search space generated by a random subset of coordinates chose at iteration t . This work can be seen as combining some of the ideas contained in works on parallel/minibatch coordinate descent [33, 30, 27] and extending them to the relatively smooth setting.

We first introduce a basic variant, which uses conservative (small) stepsizes $L_t = L$ (for $Q = \mathbb{R}^n$ this would result in stepsize $1/L$). We then perform a more detailed analysis by introducing an ESO (expected separable overapproximation) inequality [33, 30, 28] applicable to relatively smooth functions. This allows us to choose larger stepsizes $L_t \leq L$, leading to better convergence rates. In particular, under a relative strong convexity assumption (see Equations (8) and (14) for the definition), we obtain the rate (see Theorem 4.6)

$$\left(1 - p_0 \min_{i=1,2,\dots,n} \frac{v^{(i)}}{w^{(i)}}\right)^t,$$

where $p_0 = \tau/n$ is the probability that we sample any particular coordinate at each iteration, τ is the number of coordinates sampled in each iteration, $v^{(i)}$ are ESO parameters (we always have $v^{(i)} \leq L$), and $w^{(i)}$ are relative strong convexity parameters. This rate is the same as the one in [30] which applies to standard randomized coordinate descent, i.e., without relative smoothness. On the other hand, if we choose $\tau = n$, we recover relative gradient descent, and the above rate recovers the rate obtained in [3, 18].

As we show through numerical experiments, relRCD can be much faster than relGD.

Relative Stochastic Gradient Descent (relSGD). This is a special case of the generic method (7) if we choose g_t to be an unbiased estimator of $\nabla f(x_t)$, $L_t \geq L$, and $Q_t = Q$. This method extends the applicability of stochastic gradient descent to the relatively smooth setting. Convergence of the algorithm is obtained by using a specific decreasing stepsize rule (see Lemmas 5.6 and 5.7). With suitable choice of stepsizes, we obtain $O(1/t)$ convergence rate under relative strong convexity, and $O(1/\sqrt{t})$ under relative smoothness alone. The rates we obtain generalized the rates known for standard stochastic gradient descent [36].

1.5 Related work on relative smooth optimization

Relative smoothness was first introduced in [6] and later rediscovered in [3] and [18] following other works [5, 9]. In [6], Fisher market equilibrium problem was tackled and it was shown that a known algorithm to solve it, proportional response dynamics, is a special instance of relative gradient descent under relative smoothness assumption [40]. In [3] the focus is on minimizing a composite objective, $f(x) + g(x)$, where f is relatively smooth and convex, and g is convex but not necessarily differentiable. The first proximal algorithm in the relatively smooth setting is proposed there. In [18], the authors introduce the notion of relative strong convexity, and propose a dual averaging scheme. In [5], the authors show that their algorithm converges to a stationary point for nonconvex f ; no rates are given. Finally, in [9], the authors extend the ideas of dual averaging to stochastic dual averaging. However, this is only done for quadratic f .

We should also mention that there is a recent extension of minimizing relative continuous functions [17] where Lipschitzness assumption was generalized analogously as smoothness is extended by relative smoothness, opening up a new area of algorithms and applications.

1.6 Mirror Descent

Notice that the update rule (6) of relative gradient descent coincides with mirror descent update rule [21, 4]. Therefore, from practical perspective, relative gradient descent enjoys all advantages of mirror descent.

Let us now briefly review a recent mirror descent literature. We identify two main streams of work on mirror descent.

One focuses on accelerating deterministic Mirror descent using Nesterov’s idea [23]. A significant contribution in this area was done in [39], where previous methods were unified, and couple of new ones were discovered. A novel approach using the insights from ODE’s can be found in [15]. In both cases, sublinear $\mathcal{O}(k^{-2})$ rates were obtained and f was assumed to be smooth convex respectively. There is also a recent work on acceleration using coupling mirror and gradient descent [2], resulting in $\mathcal{O}(k^{-2})$ rate as well. However, to best of our knowledge, no linear rates for mirror descent are known, except of ones in the relative smooth setting.

The second stream focuses on stochastic mirror descent with access to noised gradient oracle. In [20, 10, 19] stochastic subgradient mirror descent was considered with $\mathcal{O}(k^{-1})$ rate for strongly convex and $\mathcal{O}(k^{\frac{1}{2}})$ for nonstrongly convex functions. The convergence was obtained using decreasing stepsize in this case and considering bounded variance. An accelerated stochastic mirror descent dedicated for ERM problems was proposed in [11], obtaining $\mathcal{O}(k^{-2})$ convergence rate for smooth convex but non strongly convex functions. There is also a very limited literature on coordinate mirror descent strategies. In [1], coordinate mirror descent was designed for multiple kernel learning problems. The method was casted as a special instance of stochastic mirror descent, obtaining $\mathcal{O}(k^{-1})$ convergence rate. Later in [7], stochastic block mirror descent – where the randomness appears from both coordinate choice and noised gradient was considered, obtaining $\mathcal{O}(1/k)$ rate for strongly convex and $\mathcal{O}(k^{\frac{1}{2}})$ for nonstrongly convex functions. Again, variance of the stochastic gradients was assumed to be bounded here.

To compare with our results, we stress that relative smoothness setting allows mirror map to be non-strongly convex, in contrast to virtually whole mirror descent literature. On top of that, it allows to obtain linear rates due to the (relative) strong convexity. In general, relative smooth setting allows mirror descent to be directly compared to standard gradient descent. In particular, to best of our knowledge, we develop the first stochastic mirror descent algorithm – relRCD – with linear convergence rate which outperforms relGD. The setup for relRCD is similar to randomized coordinate descent setup [30], but different to the coordinate mirror descent strategies mentioned above, as we do not consider or enforce stochastic gradient estimates, rather we take gradient descent step in batch of coordinates with stepsize determined from smoothness³. Our other contribution – relSGD – is also an extension of stochastic gradient descent in standard smooth setting. We obtain very similar rates comparing to standard mirror descent literature, however the setting we consider is different – we consider (relatively) smooth problems in contrast to [20, 10, 19], where nonsmooth problems are tackled.

2 Relatively Smooth Functions and Relative Gradient Descent

In this section, we introduce relative strongly convex property and give equivalent conditions on both relative smoothness and relative strong convexity. We also mention here a minimization algorithm under the relative smooth assumption - Relative Gradient Descent.

2.1 Relative smoothness and relative strong convexity

We firstly start by defining relative strong convexity, which is together with relative smoothness a key assumption for determining a convergence rate of algorithms mentioned in this work. Recall that we defined relative smoothness previously in (5).

³In fact, stepsize is determined from ESO assumption as in [30], which we explain in Section 4

Definition 2.1. (Relative strong convexity [18]) Function f is μ -strongly convex relative to h on Q if for any $x, y \in Q$ the following inequality holds

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \mu D_h(y, x). \quad (8)$$

As the main goal of this work is to minimize function f , we have freedom of choice of reference function h - and one would like to choose it so that the convergence rate we obtain is the best possible. In particular, as mentioned in the introduction, for a specific choice $h(x) = \frac{1}{2}\|x\|^2$ we have $D_h(x, y) = \frac{1}{2}\|x - y\|^2$ and relative strong convexity assumption becomes standard strong convexity.

The following results list some elementary properties of relative smooth functions.

Proposition 2.2 ([3, 18]). The following statements are equivalent:

- f is L -smooth relative to h on Q
- $Lh(x) - f(x)$ is a convex function on Q
- Under twice differentiability $L\nabla^2 h(x) \succcurlyeq \nabla^2 f(x)$ for all $x \in Q$
- $\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq L\langle \nabla h(x) - \nabla h(y), x - y \rangle$ for all $x \in Q$

For completeness, we also list of equivalent conditions to relative strong convexity.

Proposition 2.3 ([3, 18]). The following statements are equivalent:

- f is μ -strongly convex relative to h on Q
- $f(x) - \mu h(x)$ is a convex function on Q
- Under twice differentiability $\nabla^2 f(x) \succcurlyeq \mu \nabla^2 h(x)$ for all $x \in Q$
- $\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \mu \langle \nabla h(x) - \nabla h(y), x - y \rangle$ for all $x \in Q$

The second (convexity) and third (Hessians) conditions appearing in the two propositions above are typically easier to verify in practice. For proofs of the propositions, more properties of relatively smooth functions and some examples, we refer the reader to [3, 18].

2.2 Relative gradient descent

Now we are ready to write relative gradient descent (relGD) - baseline algorithm for minimizing relatively smooth functions.

Algorithm 1 relGD (Relative Gradient Descent) [6, 3, 18]

Input: Initial iterate x_0 ; reference function h and constant $L > 0$ such that f is L -smooth relative to h .

for $t = 0, 1, \dots, k - 1$ **do**

1. Set $x_{t+1} \leftarrow \operatorname{argmin}_{x \in Q} \{ \langle \nabla f(x_t), x \rangle + LD_h(x, x_t) \}$

end

return x_k

As mentioned in the introduction, if $h(x) = \frac{1}{2}\|x\|^2$ and $Q = \mathbb{R}^n$, we have

$$x_{t+1} = \operatorname{argmin}_{x \in Q} \left\{ \langle \nabla f(x_t), x \rangle + \frac{L}{2}\|x - x_t\|^2 \right\} = x_t - \frac{1}{L}\nabla f(x_t),$$

and relGD coincides with standard gradient descent with stepsize $\frac{1}{L}$.

Note also that Algorithm 1 is identical to Mirror descent [4]. The difference that we do not assume standard smoothness but relative smoothness with reference function h , thus the analysis and convergence results are significantly different.

The analysis of the algorithm is similar to the analysis of gradient descent under the smoothness assumption. The main difference is that one can explicitly compute the decrease in objective which is guaranteed from the standard smoothness property. This is not the case for the relative smooth optimization as we do not have a general closed expression for the next iterate. In order to overcome this issue, we are using so called three point property [16]. This is not a novel approach, it was used in [3, 18]. As we need to bound the guaranteed decrease in objective, the analysis becomes slightly looser, which is a price for the generality. However, as we show later, one can still obtain the same convergence result on the “ O ” notation comparing to the standard smooth setting.

Lemma 2.4 (Three point property). Let ϕ, h be differentiable convex functions both defined on some convex set Q . Let $D_h(\cdot, \cdot)$ be a Bregman distance. For a given $z \in Q$ denote

$$z_+ \stackrel{\text{def}}{=} \operatorname{argmin}_{x \in Q} \phi(x) + D_h(x, z).$$

Then

$$\phi(x) + D_h(x, z) \geq \phi(z_+) + D_h(z_+, z) + D_h(x, z_+), \quad \forall x \in Q. \quad (9)$$

Proof of the three point property can be found in the appendix. The following theorem states a convergence result of relative gradient descent.

Theorem 2.5 (Lu, Freund and Nesterov [18]). Consider Algorithm 1. If f is L -smooth and μ -strongly convex relative to h for some $L > 0$ and $\mu \geq 0$, then for all $k \geq 1$ the following inequality holds:

$$f(x_k) - f(x_*) \leq \frac{\mu D_h(x_*, x_0)}{\left(1 + \frac{\mu}{L-\mu}\right)^k - 1} \leq \frac{L - \mu}{k} D_h(x, x_0).$$

In the case when $\mu = 0$, the middle expression is defined in the limit as $\mu \rightarrow 0_+$.

In the case when $\mu > 0$, Relative Gradient Descent enjoys linear convergence rate, which is asymptotically driven by

$$\left(1 + \frac{\mu}{L - \mu}\right)^{-k} = \left(\frac{L}{L - \mu}\right)^{-k} = \left(1 - \frac{\mu}{L}\right)^k.$$

On the other hand if $\mu = 0$, Theorem 2.5 yields $O(1/k)$ convergence rate. Thus, relative gradient descent is, up to the constant term, matching rate of standard Gradient descent under standard smoothness assumption.

3 Relative Randomized Coordinate Descent with Short Stepsizes

In this section, we propose and analyze a naive coordinate descent algorithm for minimizing relative smooth functions. The key idea is to choose a subset of coordinates each iteration and make a step from relGD in the corresponding subspace.

We give two slightly different ways to analyze the convergence. However, neither of them provides a speedup comparing to Algorithm 1. We mention this for educational purposes, to illustrate our techniques. This issue will be addressed later in Section 4, providing us a potential speedup comparing to Algorithm 1.

The key assumption of this section - separability is defined in the following way: $h(x) = \sum_{i=1}^n h^{(i)}(x^{(i)})$, where $h^{(i)}$ takes only i -th coordinate of x . On top of that, we assume that Q is block separable: $Q = \prod_{i=1}^n Q^{(i)}$ where $Q^{(i)}$ is closed interval for all i . In other words $x \in Q$ if and only if for all i we have $x^{(i)} \in Q^{(i)}$.

Throughout this section, we assume that f is L -smooth and μ -strongly convex relative to some separable function h .

3.1 Algorithm

We introduce here Algorithm 2 – Relative Randomized Coordinate descent with short stepsizes. From now, let us denote $\mathbf{1}^i$ to be i -th column of $n \times n$ identity matrix. The update is given by (7) with

$$Q_t = \left\{ x \mid x = x_t + \sum_{i \in M_t} \text{span}(\mathbf{1}^i) \right\}.$$

Subset of coordinates M_t is chosen randomly such that $\mathbf{P}(i \in M_t) = \mathbf{P}(j \in M_t)$ for all $i, j \leq n$ and $|M_t| = \tau$.

Algorithm 2 relRCDs (Relative Randomized Coordinate Descent with Short Stepsizes)

Input: Initial iterate x_0 , separable reference function h and L such that f is L -smooth relative to h .

for $t = 0, 1, \dots, k - 1$ **do**

1. Choose $M_t \subseteq \{1, 2, \dots, n\}$ such that $\mathbf{P}(i \in M_t) = \mathbf{P}(j \in M_t)$ for all $i, j \leq n$ and $|M_t| = \tau$
2. Set $Q_t \leftarrow \{x \mid x = x_t + \sum_{i \in M_t} \text{span}(\mathbf{1}^i)\}$
3. Set $x_{t+1} \leftarrow \operatorname{argmin}_{x \in Q_t} \{\langle \nabla f(x_t), x \rangle + LD_h(x, x_t)\}$

end

return x_k

3.2 Key lemma

It will be useful to introduce

$$x_{(t+1,*)} \stackrel{\text{def}}{=} \operatorname{argmin}_{x \in Q} \{\langle \nabla f(x_t), x \rangle + LD_h(x, x_t)\}$$

as we will use this notation in the analysis.

The following lemma describes behavior of Algorithm 2 in each iteration, providing us on the expected upper bound on the value in the next iterate using the previous iterate.

Lemma 3.1 (Iteration decrease for Algorithm 2). Suppose that f is L -smooth and μ -strongly convex relative to separable function h . Then, running Algorithm 2 we obtain for all $x \in Q$:

$$\mathbf{E}[f(x_{t+1})] \leq \frac{n - \tau}{n} \mathbf{E}[f(x_t)] + \frac{\tau}{n} f(x) + \left(L - \frac{\tau}{n} \mu\right) \mathbf{E}[D_h(x, x_t)] - L \mathbf{E}[D_h(x, x_{t+1})].$$

Proof.

$$\begin{aligned} \mathbf{E}[f(x_{t+1})|x_t] &\stackrel{(5)}{\leq} f(x_t) + \mathbf{E} \left[\left(\langle \nabla f(x_t), x_{t+1} - x_t \rangle + LD_h(x_{t+1}, x_t) \right) \mid x_t \right] \\ &= f(x_t) + \mathbf{E} \left[\sum_{i \notin M_t} \left((\nabla f(x_t))^{(i)} (x_{t+1} - x_t)^{(i)} + LD_{h^{(i)}}(x_{t+1}^{(i)}, x_t^{(i)}) \right) \mid x_t \right] \\ &\quad + \mathbf{E} \left[\sum_{i \in M_t} \left((\nabla f(x_t))^{(i)} (x_{t+1} - x_t)^{(i)} + LD_{h^{(i)}}(x_{t+1}^{(i)}, x_t^{(i)}) \right) \mid x_t \right] \\ &\stackrel{(*)}{=} f(x_t) + \mathbf{E} \left[\sum_{i \in M_t} \left((\nabla f(x_t))^{(i)} (x_{t+1} - x_t)^{(i)} + LD_{h^{(i)}}(x_{t+1}^{(i)}, x_t^{(i)}) \right) \mid x_t \right] \\ &= f(x_t) + \frac{\tau}{n} \langle \nabla f(x_t), x_{(t+1,*)} - x_t \rangle + \frac{\tau}{n} LD_h(x_{(t+1,*)}, x_t) \\ &\stackrel{(9)}{\leq} f(x_t) + \frac{\tau}{n} \langle \nabla f(x_t), x - x_t \rangle + \frac{\tau}{n} LD_h(x, x_t) - \frac{\tau}{n} LD_h(x, x_{(t+1,*)}) \\ &\stackrel{(8)}{\leq} \frac{n - \tau}{n} f(x_t) + \frac{\tau}{n} f(x) + \frac{\tau}{n} (L - \mu) D_h(x, x_t) - \frac{\tau}{n} LD_h(x, x_{(t+1,*)}). \end{aligned} \tag{10}$$

The equality (*) above holds due to the fact that $x_{t+1}^{(i)} = x_t^{(i)}$ for $i \notin M_t$. Note that

$$\mathbf{E}[D_h(x, x_{t+1}) | x_t] = \frac{n-\tau}{n}D_h(x, x_t) + \frac{\tau}{n}D_h(x, x_{(t+1,*)}).$$

Plugging it into (10), we get

$$\begin{aligned} \mathbf{E}[f(x_{t+1})|x_t] &\stackrel{(10)}{\leq} \frac{n-\tau}{n}f(x_t) + \frac{\tau}{n}f(x) + \frac{\tau}{n}(L-\mu)D_h(x, x_t) - L\mathbf{E}[D_h(x, x_{t+1})|x_t] \\ &\quad + \frac{n-\tau}{n}LD_h(x, x_t) \\ &= \frac{n-\tau}{n}f(x_t) + \frac{\tau}{n}f(x) + \left(L - \frac{\tau}{n}\mu\right)D_h(x, x_t) - L\mathbf{E}[D_h(x, x_{t+1})|x_t]. \end{aligned}$$

Taking the expectation over the algorithm and using the tower property we obtain the desired result. \square

The lemma above provides us with the expected decrease in the objective every iteration. It holds for all $x \in Q$, particularly for $x = x_t$ we obtain that the sequence $\{f(x_t)\}$ is nonincreasing in expectation.

3.3 Strongly convex case: $\mu > 0$

The following theorem uses recursively Lemma 3.1 with $x = x_*$, obtaining a convergence rate of Algorithm 2.

Theorem 3.2 (Convergence rate for Algorithm 2). Suppose that f is L -smooth and μ -strongly convex relative to separable function h for $\mu > 0$. Running Algorithm 2 for k iterations we obtain:

$$\sum_{t=1}^k c_t (\mathbf{E}[f(x_t)] - f(x_*)) \leq \frac{(L - \frac{\tau}{n}\mu)D_h(x_*, x_0) + \frac{n-\tau}{n}(f(x_0) - f(x_*))}{1 - \frac{L}{\mu} + \frac{L}{\mu}\left(\frac{L}{L - \frac{\tau}{n}\mu}\right)^{k-1}},$$

where $c = (c_1, \dots, c_k) \in \mathbb{R}_+^k$ is a positive vector with entries summing up to 1.

Proof. The proof follows by applying Lemma A.1 on Lemma 3.1 with $x = x_*$ for $f_t = \mathbf{E}[f(x_t)]$, $D_t = \mathbf{E}[D_h(x_*, x_t)]$, $f_* = f(x_*)$, $\delta = \frac{\tau}{n}$, $\varphi = L$, $\psi = \mu$. \square

Note that the term driving the convergence rate in Theorem 3.2 is $(L/(L - \frac{\tau}{n}\mu))^{1-k} = (1 - \frac{\tau}{n}\frac{\mu}{L})^{k-1}$, where k is the number of iterations. In the special case when $\tau = n$, using simple algebra one can verify that Theorem 3.2 matches the results from Theorem 2.5.

3.4 Non-strongly convex case: $\mu = 0$

The following theorem provides us with the convergence rate of Algorithm 2 when f is convex but not necessarily relative strongly convex (i.e., $\mu = 0$).

Theorem 3.3 (Convergence rate for Algorithm 2). Suppose that f is convex and L -smooth relative to separable function h . Running Algorithm 2 for k iterations we obtain:

$$\sum_{t=1}^k c_t (\mathbf{E}[f(x_t)] - f(x_*)) \leq \frac{LD_h(x, x_0) + \frac{n-\tau}{n} (f(x_0) - f(x_*))}{1 + \frac{\tau(k-1)}{n}},$$

where $c = (c_1, \dots, c_k) \in \mathbb{R}^k$ is a positive vector proportional to $(\frac{\tau}{n}, \frac{\tau}{n}, \dots, \frac{\tau}{n}, 1)$.

Proof. For simplicity, denote $r_t = \mathbf{E}[f(x_t)] - f(x_*)$. We can follow the proof of Theorem 3.2 using Lemma A.1 to get the equation (35), which can be rewritten for $\mu = 0$ as follows:

$$LD_h(x, x_0) \geq r_k + \frac{\tau}{n} \sum_{t=1}^{k-1} r_t - \frac{n-\tau}{n} r_0.$$

The inequality above can be easily rearranged as

$$\frac{LD_h(x, x_0) + \frac{n-\tau}{n} r_0}{1 + (k-1)\frac{\tau}{n}} \geq \frac{1}{1 + (k-1)\frac{\tau}{n}} \left(r_k + \frac{\tau}{n} \sum_{t=1}^{k-1} r_t \right).$$

□

As previously, Theorem 3.3 captures known results of Relative Gradient Descent for $\tau = n$ (Theorem 2.5).

3.5 Improvements using a symmetry measure

For completeness, we provide a different analysis of Algorithm 2 using a different power function which is a combination of $f(x_t) - f(x_*)$ and $D_h(x_*, x_t)$. A similar analysis in the standard smooth setting was done in [38].

It would be useful to define a symmetry measure of Bregman distance here.

Definition 3.4 (Symmetry measure). Given a reference function h , the symmetry measure of D_h is defined by

$$\alpha(h) \stackrel{\text{def}}{=} \inf_{x, y} \left\{ \frac{D_h(x, y)}{D_h(y, x)} \mid x \neq y \right\}. \quad (11)$$

Note that we clearly have $0 \leq \alpha(h) \leq 1$. A symmetry measure α_h was also used in [3]. In our case, considering the symmetric measure for D_h would improve the result from the next theorem. However our results does not rely on it and hold even if there is no symmetry present, i.e. $\alpha(h) = 0$.

Theorem 3.5 (Convergence rate for Algorithm 2). Suppose that f is L -smooth and μ -strongly convex relative to separable function h . Denote $Z_t^L \stackrel{\text{def}}{=} LD_h(x_*, x_t) + f(x_t) - f(x_*)$. Running Algorithm 2 for k iterations we obtain:

$$\mathbf{E}[f(x_k) - f(x_*)] \leq \frac{Z_0^L}{1 + \frac{\tau}{n}k}$$

when $\mu = 0$ and

$$\mathbf{E} [Z_k^L] \leq \left(1 - \frac{\tau \mu}{n L} - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} \right)^k Z_0^L$$

when $\mu > 0$.

Proof. From Lemma 3.1 we have

$$\mathbf{E} [Z_{t+1}^L] \leq \mathbf{E} [Z_t^L] - \frac{\tau}{n} \mathbf{E} [Z_t^\mu]. \quad (12)$$

If $\mu = 0$, we can easily telescope the above and get the following inequality

$$\mathbf{E} [f(x_k) - f(x_*)] \leq Z_0^L - \frac{\tau}{n} k \mathbf{E} [f(x_k) - f(x_*)],$$

which leads to

$$\mathbf{E} [f(x_k) - f(x_*)] \leq \frac{Z_0^L}{1 + \frac{\tau}{n} k}.$$

Let us look at the case when $\mu \neq 0$. Firstly note that from relative strong convexity of f combining with definition of the symmetric measure $\alpha(h)$ we have

$$f(x_t) - f(x_*) \geq \mu D_h(x_t, x_*) \geq \mu \alpha(h) D_h(x_*, x_t). \quad (13)$$

Therefore, (12) can be rewritten as

$$\begin{aligned} \mathbf{E} [Z_{t+1}^L] &\stackrel{(12)}{\leq} \mathbf{E} [Z_t^L] - \frac{\tau}{n} \mathbf{E} [Z_t^\mu] \\ &= \mathbf{E} [Z_t^L] - \frac{\tau \mu}{n L} \mathbf{E} [Z_t^L] - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) (f(x_t) - f(x_*)) \\ &= \mathbf{E} [Z_t^L] - \frac{\tau \mu}{n L} \mathbf{E} [Z_t^L] - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} (f(x_t) - f(x_*)) \\ &\quad - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{L}{\mu \alpha(h) + L} (f(x_t) - f(x_*)) \\ &\stackrel{(13)}{\leq} \mathbf{E} [Z_t^L] - \frac{\tau \mu}{n L} \mathbf{E} [Z_t^L] - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} (f(x_t) - f(x_*)) \\ &\quad - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{L}{\mu \alpha(h) + L} \mu \alpha(h) D_h(x_*, x_t) \\ &= \mathbf{E} [Z_t^L] - \frac{\tau \mu}{n L} \mathbf{E} [Z_t^L] - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} \mathbf{E} [Z_t^L] \\ &= \left(1 - \frac{\tau \mu}{n L} - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} \right) \mathbf{E} [Z_t^L]. \end{aligned}$$

Using recursively the inequality above, we get

$$\mathbf{E} [Z_k^L] \leq \left(1 - \frac{\tau \mu}{n L} - \frac{\tau}{n} \left(1 - \frac{\mu}{L} \right) \frac{\mu \alpha(h)}{\mu \alpha(h) + L} \right)^k Z_0^L.$$

□

Note that as soon as $\alpha(h) = 0$, rate from the theorem above is up to the constant same as rate from Theorem 3.2 since $(L/(L - \frac{\tau}{n}\mu))^{-1} = 1 - \frac{\tau}{n}\frac{\mu}{L}$. However both theorems are measuring a convergence rate for a different quantity. On the other hand, in the best case if $\alpha(h) = 1$ we have

$$1 - \frac{\tau}{n}\frac{\mu}{L} - \frac{\tau}{n}\left(1 - \frac{\mu}{L}\right)\frac{\mu}{\mu + L} = 1 - \frac{\tau}{n}\frac{\mu}{L} - \frac{\tau}{n}\frac{\mu}{L}\left(1 - \frac{2\mu}{L + \mu}\right) \geq 1 - 2\frac{\tau}{n}\frac{\mu}{L},$$

thus the convergence rate we obtained might be up to 2 times faster comparing to rate from Theorem 3.2. Thus the convergence rate is also up to 2 times faster comparing to Theorem 2.5 for the case $\tau = n$ if $\alpha(h) < 0$. On the other hand, Theorem 3.5 provides us with convergence rate of $\mathbf{E}[D_h(x_*, x_k)]$, as the following inequality trivially holds:

$$\mathbf{E}[D_h(x_*, x_k)] \leq \frac{\mathbf{E}[Z_k^L]}{L}.$$

Suppose that we have a fixed budget on the total work of the algorithm, i.e. we can make only k/τ iterations. It is a simple exercise to notice that the bound on the suboptimality for Theorems 3.2, 3.3 and 3.5 after k/τ iterations is not getting better when minibatch size τ is decreasing. We address next section in order to solve this issue.

4 Relative Randomized Coordinate Descent with Large Stepsizes

This section addresses the issue of the previous section - allowing a better usage of randomness in order to obtain a faster convergence rate comparing to the deterministic setting. Theorem 4.6 later in this section is one of two key results (together with the analysis of Relative Stochastic Gradient Descent) of this work.

As previously, we assume that h is separable function, i.e., $h(x) = \sum_{i=1}^n h^{(i)}(x^{(i)})$ and Q is block separable $Q = \prod_{i=1}^n Q^{(i)}$. For notational simplicity, let us define weighted Bregman distance and weighted inner product:

$$D_h(x, y)_v \stackrel{\text{def}}{=} \sum_{i=1}^n v^{(i)} \left(h^{(i)}(x^{(i)}) - h^{(i)}(y^{(i)}) - \nabla h^{(i)}(y^{(i)}) \cdot (x^{(i)} - y^{(i)}) \right),$$

$$\langle a, b \rangle_p \stackrel{\text{def}}{=} \sum_{i=1}^n p_i a_i b_i,$$

where $v, p \in \mathbb{R}^n$ are some positive vectors.

It would be also useful to introduce the separable version of relative strong convexity, as a generalization of Relative Strong Convexity with respect to a separable function h , allowing different strong convexity parameters for each coordinate.

Definition 4.1 (Relative strong convexity, separable version). Suppose that $w \in \mathbb{R}_+^n$. Function f is w -strongly convex relative to separable function h on Q if for any $x, y \in \text{int}(Q)$ the following inequality holds

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + D_h(y, x)_w. \quad (14)$$

Throughout this section, we will assume the separable version of relative strong convexity, as it captures relative μ -strong convexity as a special case for $w = \mu \mathbf{1}$ and might potentially bring a better convergence result.

4.1 Expected separable overapproximation

In our analysis, we use h -ESO assumption defined below instead of relative smoothness assumption. In the standard smoothness setting, it was firstly introduced in [31].

Definition 4.2 (h -ESO). Let h be some separable function and $p = p(\hat{S})$ be a probability vector of the sampling \hat{S} , i.e. $p^{(i)} = \mathbf{P}(i \in \hat{S})$. Function f admits Expected Separable Overapproximation with respect to function h (h -ESO), parameters \hat{S} (probability sampling) and v (vector) if the following inequality holds for all $x, q \in \mathbb{R}^n$:

$$\mathbf{E} \left[f \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right) \right] \leq f(x) + \langle \nabla f(x), q \rangle_p + D_h(x + q, x)_{p \circ v}. \quad (15)$$

For simplicity, we write $(f, \hat{S}) \sim \text{ESO}_h(v)$.

Above, “ \circ ” denotes Hadamard product, i.e. element-wise product of two vectors. Note that if f is L -smooth relative to the separable function h , then we have

$$\begin{aligned} \mathbf{E} \left[f \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right) \right] &\leq \mathbf{E} \left[f(x) + \left\langle \nabla f(x), \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right\rangle + LD_h \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i, x \right) \right] \\ &= f(x) + \langle \nabla f(x), q \rangle_p + D_h(x + q, x)_{pL} \end{aligned}$$

and thus $(f, \hat{S}) \sim \text{ESO}_h(L\mathbf{1})$. In other words, if f is L -smooth relative to separable function h , then $(f, \hat{S}) \sim \text{ESO}_h(L\mathbf{1})$ with any sampling \hat{S} .

However, when considering a specific sampling strategy, it might be possible to choose smaller ESO parameters v , allowing us to obtain a faster convergence comparing to the deterministic method using full gradient in each iteration. As we show later, if ESO parameters are chosen to be smoothness parameters, we do not obtain any speedup comparing the deterministic method.

There are various examples of functions satisfying h -ESO:

- If $h(x) = \|x\|^2/2$, definition of h -ESO matches definition of standard Expected Separable Overapproximation introduced in [28], [31]. Under the assumption that f is $A^\top A$ smooth, i.e. $\forall x, y$:

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2}(y - x)^\top A^\top A(y - x),$$

one can prove that $(f, \hat{S}) \sim \text{ESO}_h(v)$ if

$$P(\hat{S}) \circ (A^\top A) \preceq \text{Diag} \left(p(\hat{S}) \circ v \right),$$

where $P(\hat{S})$ and $p(\hat{S})$ are respectively probability matrix and probability vector of sampling \hat{S} .

Note that $A^\top A$ smoothness is equivalent to relative smoothness for $L = 1$, $h(x) = \frac{1}{2}x^\top A^\top A x$ and arises naturally if the objective f is in the form

$$f(x) \stackrel{\text{def}}{=} \sum_{i=1}^n \phi^{(i)}(M_{(i)}x),$$

where function $\phi^{(i)}$ is $\gamma^{(i)}$ smooth. In this case, f is $A^\top A \stackrel{\text{def}}{=} \sum_{i=1}^n \gamma^{(i)} M_{(i)}^\top M_{(i)}$ smooth. As an example, for uniform sampling (when every iteration is only one coordinate sampled uniformly at random), v can be chosen as $\text{Diag}(A^\top A)$. In contrast, the tightest smoothness parameter that can be chosen here is the maximal eigenvalue of $(A^\top A)$, which is in general even greater than maximal diagonal element of $(A^\top A)$.

For more details about how to choose v for arbitrary sampling \hat{S} or proofs of the statements above, see [28].

- D-optimal design problem.

$$\begin{aligned} \min_x \quad & f(x) \stackrel{\text{def}}{=} \log \det \left(H \text{Diag}(x) H^\top \right) \\ \text{subject to} \quad & \langle \mathbf{1}, x \rangle = 1 \\ & x \in \mathbb{R}_+^n, \end{aligned}$$

where matrix $H \in \mathbb{R}^{m \times n}$ has rank n , $n \geq m + 1$. In this case f is 1 relative smooth with respect to $h(x) \stackrel{\text{def}}{=} -\sum_{i=1}^n \log(x^{(i)})$ [18]. Thus, function $(f, \hat{S}) \sim \text{ESO}_h(\mathbf{1})$ for any sampling \hat{S} .

- Poisson linear inverse problem. The task here is to find vector $x \in \mathbb{R}_+^n$ to minimize $\text{KL}(Ax||b)$ for matrix $A \in \mathbb{R}_+^{m \times n}$ and vector $b \in \mathbb{R}_+^m$.

Thus optimization problem here is the following:

$$\begin{aligned} \min_x \quad & f(x) \stackrel{\text{def}}{=} \sum_{i=1}^m f^{(i)}(x) = \sum_{i=1}^m \left(b^{(i)} \log \frac{b^{(i)}}{(Ax)^{(i)}} + (Ax)^{(i)} - b^{(i)} \right) \\ \text{subject to} \quad & x \in \mathbb{R}_+^n. \end{aligned}$$

Again, in this case f is $\sum_{i=1}^m b^{(i)}$ -smooth relative to $h(x) \stackrel{\text{def}}{=} -\sum_{i=1}^n \log(x^{(i)})$ [3]. Thus, as before, $(f, \hat{S}) \sim \text{ESO}_h(\mathbf{1} \sum_{i=1}^m b^{(i)})$ for any sampling \hat{S} .

Considering regularized Poisson linear inverse problem:

$$\min_x \quad f(x) \stackrel{\text{def}}{=} \text{KL}(Ax||b) + \mu r(x)$$

with logarithmic regularizer $r(x) = -\sum_{i=1}^m \log(x^{(i)})$. Then we have

$$(f, \hat{S}) \sim \text{ESO}_h \left(\left(\sum_{i=1}^m b^{(i)} + \mu \right) \mathbf{1} \right)$$

for any sampling \hat{S} .

The following lemma gives an example on function f which is h -ESO where h is not $\frac{1}{2}\|x\|^2$ with parameters v potentially n times smaller than relative smoothness constant L .

Lemma 4.3. Suppose that

$$f(x) \stackrel{\text{def}}{=} f_1(x) + f_2(x),$$

where f_1 is L_1 smooth relative to h_1 and f_2 is $A^\top A$ smooth (1-smooth relative to $h_2(x) \stackrel{\text{def}}{=} \frac{1}{2}x^\top A^\top A x$). Let us consider \hat{S} to be uniform sampling which samples a single coordiante (uniformly)

each iteration. Then,

$$(f, \hat{S}) \sim \text{ESO}_h \left(\max \left(L_1 \mathbf{1}, \text{diag}(A^\top A) \right) \right)$$

for

$$h(x) \stackrel{\text{def}}{=} h_1(x) + \frac{1}{2} \|x\|^2.$$

Proof. From ESO theory in standard smooth setting we have that $(f_2, \hat{S}) \sim \text{ESO}_{h_2}(\text{diag}(A^\top A))$ for $h_2(x) \stackrel{\text{def}}{=} \frac{1}{2} \|x\|^2$. Clearly, $(f_1, \hat{S}) \sim \text{ESO}_{h_1}(L_1 \mathbf{1})$. Summing ESO inequality for f_1 and f_2 we get

$$\begin{aligned} \mathbf{E} \left[f \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right) \right] &= \mathbf{E} \left[f_1 \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right) + f_2 \left(x + \sum_{i \in \hat{S}} q^{(i)} \mathbf{1}^i \right) \right] \\ &\leq f_1(x) + \langle \nabla f_1(x), q \rangle_p + D_{h_1}(x + q, x)_{p \circ L_1 \mathbf{1}} \\ &\quad + f_2(x) + \langle \nabla f_2(x), q \rangle_p + \frac{1}{2} \|q\|_{p \circ v}^2 \\ &\leq f(x) + \langle \nabla f(x), q \rangle_p + D_h(x + q, x)_{p \circ v}, \end{aligned}$$

which concludes the proof. □

Note that in the lemma above f is L -smooth relative to h where $L \stackrel{\text{def}}{=} \max(L_1, \lambda_{\max}(A^\top A))$, and in general one cannot find tighter constant L . Clearly, $v^{(i)} \leq L$ for all i and in the case when $A = \mathbf{1}$ and $L_1 < 1$ we have $\lambda_{\max}(A^\top A) = n$ and thus $L = n$, in contrast to $v = \mathbf{1}$, thus L might be n times larger than ESO parameters v . We also note without the proof that it is possible to design ESO parameters for block coordinate descent as well analogously.

4.2 Algorithm

Let us now proceed with the algorithm. We introduce here Relative Randomized Coordinate Descent (relRCD) - algorithm for minimizing functions satisfying Relative ESO assumption. The main idea is very simple - each iteration sample a subset of coordinates with respect to sampling \hat{S} and update them according to Relative ESO assumption. We only consider sampling strategies such that all coordinates have equal chance to be sampled.

Algorithm 3 relRCD (Relative Randomized Coordinate Descent)

Input: Initial iterate x_0 , separable reference function h , positive vector v and sampling $\hat{S} (f, \hat{S}) \sim \text{ESO}_h(v)$ and $\mathbf{P}(i \in \hat{S}) = \mathbf{P}(j \in \hat{S})$ for all $i, j \leq n$.

for $t = 0, 1, \dots, k - 1$ **do**

1. Choose randomly $M_t \in \{1, 2, \dots, m\}$ according to the sampling \hat{S}
2. Set $Q_t \leftarrow \left\{ x \mid x = x_t + \sum_{i \in M_t} \text{span}(\mathbf{1}^i) \right\}$
3. Set $x_{t+1} \leftarrow \text{argmin}_{x \in Q_t} \langle \nabla f(x_t), x \rangle + D_h(x, x_t)_v$

end

return x_k

4.3 Analysis

First of all, we introduce the variant of three point property, which we will use later in the analysis. For simplicity we denote probability vector of sampling \hat{S} as p throughout this section. Since all coordinates have the same probability to be sampled, we can write $p = p_0 \mathbf{1}$ for some scalar p_0 such that $0 < p_0 \leq 1$.

Lemma 4.4 (Three point property for ESO). Let $c, v, p \in \mathbb{R}^n$ and $D_h(\cdot, \cdot)$ be a Bregman distance for separable function $h(x) = \sum_{i=1}^n h^{(i)}(x)$, both defined on some arbitrary set Q . For a given $z \in Q$ denote

$$z_+ \stackrel{\text{def}}{=} \text{argmin}_{x \in Q} \{ \langle c, x \rangle_p + D_h(x, z)_{p \circ v} \},$$

where $D_h(x, z)_{p \circ v} = \sum_{i=1}^n D_{h^{(i)}}(x^{(i)}, z^{(i)}) p^{(i)} v^{(i)}$. Then for all $x \in Q$ we have

$$\langle c, x \rangle_p + D_h(x, z)_{p \circ v} \geq \langle c, z_+ \rangle_p + D_h(z_+, z)_{p \circ v} + D_h(x, z_+)_{p \circ v}. \quad (16)$$

Proof. Define $c' = c \circ p$ and $h'(x) = \sum_{i=1}^n p^{(i)} v^{(i)} h^{(i)}(x^{(i)})$. Thus we have

$$z_+ = \text{argmin}_{x \in Q} \{ \langle c', x \rangle + D_{h'}(x, z) \}.$$

It remains to apply the three point property (Lemma 2.4). □

The next lemma provides us with the expected decrease in objective for each iteration of Algorithm 3 and has the same role as Lemma 3.1 in the analysis if Algorithm 2.

For notational simplicity, denote throughout this section

$$x_{(t+1,*)} \stackrel{\text{def}}{=} \text{argmin}_{x \in Q} \langle \nabla f(x_t), x \rangle + D_h(x, x_t)_v. \quad (17)$$

Lemma 4.5 (Iteration decrease for Algorithm 3). Suppose that f is w -Relative Strongly Convex with respect to h and $(f, \hat{S}) \sim \text{ESO}_h(v)$ for $p(\hat{S}) = p = p_0 \mathbf{1}$. Denote $\Delta = \min \frac{w^{(i)}}{v^{(i)}}$. Then, one iteration of relRCD satisfies

$$\mathbf{E}[f(x_{t+1})] \leq (1 - p_0) \mathbf{E}[f(x_t)] + p_0 f(x_*) + (1 - p_0 \Delta) \mathbf{E}[D_h(x_*, x_t)_v] - \mathbf{E}[D_h(x_*, x_{t+1})_v].$$

Proof. Let us write h -ESO for $x = x_t$, $q = x_{(t+1,*)} - x_t$ and sampling \hat{S} . We get

$$\begin{aligned}
\mathbf{E}[f(x_{t+1}) | x_t] &\stackrel{(15)}{\leq} f(x_t) + \langle \nabla f(x_t), x_{(t+1,*)} - x_t \rangle_p + D_h(x_{(t+1,*)}, x_t)_{pov} \\
&\stackrel{(16)}{\leq} f(x_t) + \langle \nabla f(x_t), x - x_t \rangle_p + D_h(x, x_t)_{pov} - D_h(x, x_{(t+1,*)})_{pov} \\
&\stackrel{(14)}{\leq} (1 - p_0)f(x_t) + p_0f(x) - D_h(x, x_t)_{pov} + D_h(x, x_t)_{pov} - D_h(x, x_{(t+1,*)})_{pov} \\
&\leq (1 - p_0)f(x_t) + p_0f(x) + (1 - \Delta)D_h(x, x_t)_{pov} - D_h(x, x_{(t+1,*)})_{pov}. \quad (18)
\end{aligned}$$

In the last inequality above we used the definition of Δ . Since

$$\mathbf{E}[D_h(x, x_{t+1})_v | x_t] = (1 - p_0)D_h(x, x_t)_v + p_0D_h(x, x_{(t+1,*)})_v,$$

we have

$$D_h(x, x_{(t+1,*)})_{pov} = \mathbf{E}[D_h(x, x_{t+1})_v | x_t] - (1 - p_0)D_h(x, x_t)_v.$$

Plugging it back to (18) we obtain

$$\begin{aligned}
\mathbf{E}[f(x_{t+1}) | x_t] &\leq (1 - p_0)f(x_t) + p_0f(x) + (1 - \Delta)D_h(x, x_t)_{pov} - \mathbf{E}[D_h(x, x_{t+1}) | x_t] \\
&\quad + (1 - p_0)D_h(x, x_t)_v \\
&= (1 - p_0)f(x_t) + p_0f(x) + (1 - p_0\Delta)D_h(x, x_t)_v - \mathbf{E}[D_h(x, x_{t+1})_v | x_t]. \quad (19)
\end{aligned}$$

Taking the expectation over the algorithm and using the tower property we obtain the desired result. \square

Now, we are ready to introduce first of the two main results of this work - Theorems 4.6 and 4.7, providing with a convergence rate of relRCD under ESO assumption.

4.3.1 Strongly convex case $w \in \mathbb{R}_{++}^n$

Theorem 4.6 (Convergence rate for Algorithm 3). Suppose that f is w -strongly convex relative to h for $w \in \mathbb{R}_+^n$ and that $(f, \hat{S}) \sim \text{ESO}_h(v)$ for $p(\hat{S}) = p = p_0\mathbf{1}$. Denote $\Delta = \min \frac{w^{(i)}}{v^{(i)}}$. Then, iterates of Algorithm 3 satisfy:

$$\sum_{t=1}^k c_t (\mathbf{E}[f(x_t)] - f(x_*)) \leq \frac{(1 - p_0\Delta)D_h(x_*, x_0)_v + (1 - p_0)(f(x_0) - f(x_*))}{1 - \Delta^{-1} + \Delta^{-1} \left(\frac{1}{1 - p_0\Delta} \right)^{k-1}}, \quad (20)$$

where $c \in \mathbb{R}^k$ is a positive vector with entries summing up to 1. On top of that, we have

$$\mathbf{E}[D_h(x_*, x_k)_v] \leq (1 - p_0\Delta)^k D_h(x_*, x_0)_v, \quad (21)$$

and

$$\frac{1}{k} \sum_{t=1}^k \mathbf{E} [D_h(x_t, x_{(t+1,*)})_v] \leq \frac{f(x_0) - f(x_*)}{kp_0}. \quad (22)$$

Proof. The proof of (20) follows by applying Lemma A.1 together with Lemma 4.5 for $f_t = \mathbf{E} [f(x_t)]$, $D_t = \mathbf{E} [D_h(x_*, x_t)_v]$, $f_* = f(x_*)$, $\delta = p_0$, $\varphi = 1$, $\psi = \Delta$.

Inequality (21) follows recursively from

$$\mathbf{E} [D_h(x_*, x_{t+1})_v] \leq (1 - p_0\Delta) \mathbf{E} [D_h(x_*, x_t)_v],$$

which holds due to Lemma 4.5 as $\mathbf{E} [f(x_t)]$ is a nonincreasing sequence.

Finally, to prove inequality (22) let us set $x = x_t$ in (19) to obtain

$$\mathbf{E} [f(x_{t+1}) | x_t] \leq f(x_t) - \mathbf{E} [D_h(x_t, x_{t+1})_v | x_t].$$

Taking the full expectation, averaging over iterations and using $\mathbf{E} [f(x_k)] \leq f(x_*)$ we get

$$\frac{1}{k} \sum_{t=1}^k \mathbf{E} [\mathbf{E} [D_h(x_t, x_{t+1})_v | x_t]] \leq \frac{f(x_0) - f(x_*)}{k}.$$

It remains to notice that $\mathbf{E} [D(x_t, x_{t+1}) | x_t] = p_0 D(x_t, x_{(t+1,*)})$. □

Convergence rates from (20) and (21) are both asymptotically driven by the term

$$(1 - p_0\Delta)^k = \left(1 - p_0 \min_i \frac{v^{(i)}}{w^{(i)}} \right)^k.$$

Therefore, no speedup is obtained comparing to relGD (Theorem 2.5), for ESO parameters set as $v = L\mathbf{1}$ and strong convexity parameters set as $w = \mu\mathbf{1}$. However, if one set ESO parameters v more tightly, taking into the consideration the specific probability sampling, one can outperform Algorithm 1. There is a broad theory about how to compute ESO parameters v for various different sampling strategies in case of $h(x) = \frac{1}{2}\|x\|^2$, see [28]. We gave the example of one class of functions in Lemma 4.3.

Note also that (20) provides an asymptotically same convergence result as Randomized Coordinate Descent in the standard smooth setting for uniform sampling [30], therefore we obtained a good generalization in this case.

To conclude this section, notice that (22) provides us with a convergence of $\mathbf{E} [D_h(x_t, x_{t+1})_v]$. Quantity $D_h(x_t, x_{t+1})_v$ depends on x_t , h and f and goes to 0 when $\nabla f(x_t)$ goes to 0 (this can be easily seen from (17)). Thus $D_h(x_t, x_{t+1})_v$ can be considered as a “norm” of $\nabla f(x_t)$ which depends on x_t and h . In the standard setting when $h(x) = \|x\|^2/2$ and $v = L\mathbf{1}$ we have

$$D_h(x_t, x_{t+1})_v = LD_h(x_t, x_{t+1}) = L \left\| \frac{1}{L} \nabla f(x_k) \right\|^2 = \frac{1}{L} \|\nabla f(x_k)\|^2,$$

and thus we obtain the convergence of the norm of gradient in this case.

Remark 1. According to Theorem 4.6, one needs

$$\frac{\Delta}{p_0} \log(\Delta) \log \left(\frac{(1 - p_0\Delta)D_h(x_*, x_0)_v + (1 - p_0)(f(x_0) - f(x_*))}{\epsilon} + \Delta^{-1} - 1 \right) \quad (23)$$

iterations for Algorithm 3 to converge to ϵ -optimality in functional values and

$$\frac{\Delta}{p_0} \log \left(\frac{D_h(x_*, x_0)_v}{\epsilon} \right) \quad (24)$$

iterations to get to ϵ -neighborhood to the optimum in (Bregman) distance. For a comparison, randomized coordinate descent in standard smooth setting requires

$$\frac{\Delta}{p_0} \log \left(\frac{f(x_0) - f(x_*)}{\epsilon} \right)$$

iterations to reach to ϵ -optimality, which is essentially same as both (23) and (24).

4.3.2 Non-strongly convex case: $\min w_i = 0$

The following theorem provides us with the convergence rate of Algorithm 3 when f is convex but not necessarily relative strongly convex (i.e., $\min w_i = 0$).

Theorem 4.7 (Convergence rate for Algorithm 3). Suppose that f is convex and $(f, \hat{S}) \sim \text{ESO}_h(v)$ for $p(\hat{S}) = p = p_0 \mathbf{1}$ and separable convex function h . Running Algorithm 3 for k iterations we obtain:

$$\sum_{t=1}^k c_t (\mathbf{E}[f(x_t)] - f(x_*)) \leq \frac{D_h(x, x_0)_v + (1 - p_0)(f(x_0) - f(x_*))}{1 + p_0(k - 1)},$$

where $c = (c_1, \dots, c_k) \in \mathbb{R}^k$ is a positive vector proportional to $(p_0, p_0, \dots, p_0, 1)$.

Proof. For simplicity, denote $r_t = \mathbf{E}[f(x_t)] - f(x_*)$. We can follow the proof of Theorem 3.2 using Lemma A.1 to get the equation (35), which can be rewritten for $\mu = 0$ as follows:

$$D_h(x, x_0)_v \geq r_k + p_0 \sum_{t=1}^{k-1} r_t - (1 - p_0)r_0,$$

which can be easily rearranged as

$$\frac{D_h(x, x_0)_v + (1 - p_0)r_0}{1 + (k - 1)p_0} \geq \frac{1}{1 + (k - 1)p_0} \left(r_k + p_0 \sum_{t=1}^{k-1} r_t \right).$$

□

As previously, Theorem 3.3 captures known results of Relative Gradient Descent for $p_0 = 1$ (Theorem 2.5).

5 Relative Stochastic Gradient Descent

In this section, we assume that every iteration we have an access to the stochastic oracle providing us \tilde{g}_t – an unbiased estimator of $\nabla f(x_t)$. The next iterate of the algorithm is obtained using the stochastic gradient instead of the true gradient. The analogous algorithm in the standard smooth setting is Stochastic Gradient Descent which is in fact a special case of Relative Stochastic Gradient Descent.

5.1 Algorithm

The iterates of standard stochastic gradient descent with stepsize sequence $\{\gamma_t\}_{t=0}^{\infty}$ are the following

$$x_{t+1} \leftarrow x_t - \gamma_t \tilde{g}_t. \quad (25)$$

It is known that unlike gradient descent, scheme (25) does not necessarily guarantee the convergence to the optimum, as the variance of gradient estimator \tilde{g}_t might not converge to zero, resulting in the convergence to the neighborhood of the optimum. This is where the importance of decreasing stepsize sequence $\{\gamma_t\}_{t=0}^{\infty}$ takes a place; thus taking more conservative steps as progressing with the algorithm. However, in particular special cases, such as empirical risk minimization, a different tricks tricks guarantee vanishing variance of gradient estimator as one approach optimum [35, 8, 12, 37, 25].

In this work, we attain the convergence of Relative Stochastic Gradient Descent by making the algorithm conservative over time. We leave the variance reduction for relatively smooth ERM problems as an open research question.

Algorithm 4 relSGD (Relative Stochastic Gradient Descent)

Input: Initial iterate x_0 , separable reference function h , positive scalar L such that f is L -relative smooth with respect to h , stepsize determining sequence $\{L_t\}_{t=0}^{\infty}$ with $L_0 = L$.

for $t = 0, 1, \dots, k - 1$ **do**

1. Get \tilde{g}_t such that $\mathbf{E}[\tilde{g}_t] = \nabla f(x_t)$
2. Set $x_{t+1} \leftarrow \operatorname{argmin}_{x \in Q} \{\langle \tilde{g}_t, x \rangle + L_t D_h(x, x_t)\}$

end

return x_k

Recall that for the special choice $D_h(x, y) = \frac{1}{2}\|x - y\|^2$, Stochastic Gradient Descent with nonincreasing stepsize $\gamma_t = \frac{1}{L_t}$ is recovered. Define the new iterate using the true gradient as

$$x_{(t+1,*)} \stackrel{\text{def}}{=} \operatorname{argmin}_{x \in Q} \{\langle \nabla f(x_t), x \rangle + L_t D_h(x, x_t)\},$$

which will be used only in Assumption 5.1, and will never be evaluated in the actual run of the algorithm.

Throughout this section, we will make the following assumption, which is in fact closely related to boundedness of variance of the gradient estimator, as Remark 2 shows. Notice that boundedness of variance of gradient estimator is very common in SGD literature.

Assumption 5.1. There exist $\sigma \neq 0$ such that for all t we have

$$L_t \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_{(t+1,*)} \rangle \mid x_t] \leq \sigma^2. \quad (26)$$

Remark 2. Consider Assumption 5.1. If we additionally assume that h is μ_h -strongly convex function, we obtain

$$\begin{aligned}
L_t \mathbf{E} [\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_{(t+1,*)} \rangle \mid x_t] &\leq L_t \mathbf{E} \left[\|\nabla f(x_t) - \tilde{g}_t\| \cdot \|x_{t+1} - x_{(t+1,*)}\| \mid x_t \right] \\
&\stackrel{(*)}{\leq} L_t \mathbf{E} \left[\|\nabla f(x_t) - \tilde{g}_t\| \cdot \frac{1}{\mu_h} \left\| \frac{1}{L_t} (\tilde{g}_t - \nabla f(x_t)) \right\| \mid x_t \right] \\
&= \frac{1}{\mu_h} \mathbf{E} \left[\|\nabla f(x_t) - \tilde{g}_t\|^2 \mid x_t \right].
\end{aligned}$$

Inequality (*) holds due to μ_h -strong convexity of h , since

$$\|x_{t+1} - x_{(t+1,*)}\| \leq \frac{1}{\mu_h} \|\nabla h(x_{t+1}) - \nabla h(x_{(t+1,*)})\| = \frac{1}{\mu_h} \left\| \frac{1}{L_t} (\tilde{g}_t - \nabla f(x_t)) \right\|.$$

Thus, if h is μ_h -strongly convex, σ^2 can be chosen so that $\sigma^2 \mu_h$ correspond to the global upper bound on variance of gradient estimator.

5.2 Key Lemma

The following lemma is key for this section and provides us with a bound on expected suboptimality in iteration t .

Lemma 5.2 (Iteration decrease for Algorithm 4). Suppose that f is L -smooth and μ -strongly convex relative to function h . Performing one iteration of Algorithm 4 we obtain for all $x \in Q$

$$\begin{aligned}
\mathbf{E} [f(x_{t+1}) \mid x_t] - f(x) &\leq (L_t - \mu) D_h(x, x_t) - L_t \mathbf{E} [D_h(x, x_{t+1}) \mid x_t] \\
&\quad + \frac{\sigma^2}{L_t} - (L_t - L) \mathbf{E} [D_h(x_{t+1}, x_t) \mid x_t]. \tag{27}
\end{aligned}$$

Proof.

$$\begin{aligned}
\mathbf{E}[f(x_{t+1}) | x_t] &\stackrel{(5)}{\leq} f(x_t) + \mathbf{E}[\langle \nabla f(x_t), x_{t+1} - x_t \rangle + LD_h(x_{t+1}, x_t) | x_t] \\
&= f(x_t) + \mathbf{E}[\langle \nabla f(x_t), x_{t+1} - x_t \rangle + L_t D_h(x_{t+1}, x_t)] \\
&\quad - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&= f(x_t) + \mathbf{E}[\langle \nabla f(x_t), x_{t+1} - x_t \rangle + L_t D_h(x_{t+1}, x_t) | x_t] \\
&\quad - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&= f(x_t) + \mathbf{E}[\langle \tilde{g}_t, x_{t+1} - x_t \rangle + L_t D_h(x_{t+1}, x_t) | x_t] \\
&\quad + \mathbf{E}[\langle \nabla f(x_t), x_{t+1} - x_t \rangle - \langle \tilde{g}_t, x_{t+1} - x_t \rangle | x_t] \\
&\quad - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&\stackrel{(9)}{\leq} f(x_t) + \mathbf{E}[\langle \tilde{g}_t, x - x_t \rangle + L_t D_h(x, x_t) - L_t D_h(x, x_{t+1}) | x_t] \\
&\quad + \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_t \rangle | x_t] - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&= f(x_t) + \langle \nabla f(x_t), x - x_t \rangle + L_t D_h(x, x_t) - L_t \mathbf{E}[D_h(x, x_{t+1}) | x_t] \\
&\quad + \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_t \rangle | x_t] - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&\stackrel{(8)}{\leq} f(x) + (L_t - \mu) D_h(x, x_t) - L_t \mathbf{E}[D_h(x, x_{t+1}) | x_t] \\
&\quad + \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_t \rangle | x_t] - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&\stackrel{(*)}{=} f(x) + (L_t - \mu) D_h(x, x_t) - L_t \mathbf{E}[D_h(x, x_{t+1}) | x_t] \\
&\quad + \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_{(t+1,*)} \rangle | x_t] - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t] \\
&\stackrel{(26)}{\leq} f(x) + (L_t - \mu) D_h(x, x_t) - L_t \mathbf{E}[D_h(x, x_{t+1}) | x_t] + \frac{\sigma^2}{L_t} \\
&\quad - (L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t].
\end{aligned}$$

Equality (*) follows from fact that \tilde{g}_t is unbiased and thus we have

$$\mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_t \rangle | x_t] = \mathbf{E}[\langle \nabla f(x_t) - \tilde{g}_t, x_{(t+1,*)} \rangle | x_t] = 0.$$

□

Note that Lemma 5.2 is very similar to Lemma 3.1 for $\tau = n$. There are only two additional terms in (27) – $\frac{\sigma^2}{L_t}$ appears due to the noise in the gradient estimator and $(L_t - L) \mathbf{E}[D_h(x_{t+1}, x_t) | x_t]$ appears due to the varying stepsize rule. We now derive the convergence rate of relSGD for various stepsize rules.

5.3 Constant stepsize rule

The following theorem provides a convergence result of SGD with constant stepsize rule using recursively Lemma 5.2 – it shows that Relative Stochastic Gradient Descent converges linearly to a particular neighborhood of the optimum. We mention it for completeness, to illustrate that relSGD in our fully general relative smooth setting behaves very similar to standard (smooth) SGD.

Theorem 5.3 (Constant stepsize rule for Algorithm 4). Suppose that f is L -smooth and μ -strongly convex relative to h . Iterates of Algorithm 4 with stepsize rule $L_t = L$ satisfy:

$$\sum_{t=1}^k c_t \left(\mathbf{E}[f(x_t)] - \left(f(x_*) + \frac{\sigma^2}{L} \right) \right) \leq \frac{D_h(x_*, x_0) \mu}{\left(\frac{L}{L-\mu} \right)^k - 1}, \quad (28)$$

where c is positive vector proportional to $(1, \beta, \beta^2, \dots, \beta^{k-1})$ summing up to 1 for

$$\beta \stackrel{\text{def}}{=} \frac{L}{L - \frac{\mu}{m}}.$$

Proof. Let us set $x = x_*$ in (27), take the expectation of over the algorithm and use the tower property. We obtain

$$\mathbf{E}[f(x_{t+1})] - \left(f(x_*) + \frac{\sigma^2}{L} \right) \leq (L - \mu) \mathbf{E}[D_h(x_*, x_t)] - L \mathbf{E}[D_h(x_*, x_{t+1})].$$

The proof now follows directly by applying Lemma A.1 the inequality above for $f_t = \mathbf{E}[f(x_t)]$, $D_t = \mathbf{E}[D_h(x_*, x_t)]$, $f_* = f(x_*) + \frac{\sigma^2}{L}$, $\delta = 1$, $\varphi = L$, $\psi = \mu$. \square

Inequality (28) shows that the sequence of iterates $\{x_t\}$ converges linearly to the set $\{x : f(x) \leq f(x_*) + \frac{\sigma^2}{L}\}$, and the convergence rate is driven by the term $(1 - \frac{\mu}{L})^k$.

5.4 Decreasing stepsize rule

The following theorem is one of two key results of this work, together with Theorem 4.6. It provides us with a convergence result of Algorithm 4 for a general stepsize rule.

Theorem 5.4 (General convergence for Algorithm 4). Suppose that f is L -smooth and μ -strongly convex relative to h . Define $c_0 = 1$ and $c_t = \frac{L_{t-1}}{L_t - \mu} c_{t-1}$ for $t \geq 1$ and $C_k = \sum_{t=1}^k c_{t-1}$. Then, Algorithm 4 satisfies:

$$\sum_{t=1}^k \frac{c_{t-1}}{C_k} \mathbf{E}[f(x_t) - f(x_*)] \leq \frac{(L - \mu) D_h(x_*, x_0)}{C_k} + \sigma^2 \sum_{t=0}^{k-1} \frac{c_t}{C_k L_t}. \quad (29)$$

Proof. Let us set $x = x_t$ in (27), take the expectation of over the algorithm and use tower property. Ignoring the last term we get

$$\mathbf{E}[f(x_{t+1}) - f(x_*)] \leq (L_t - \mu) D_h(x_*, x_t) - L_t \mathbf{E}[D_h(x_*, x_{t+1})] + \frac{\sigma^2}{L_t}.$$

Multiplying the above by c_t and summing for $t = 0$ to $k - 1$ we obtain

$$\begin{aligned}
\sum_{t=1}^k c_{t-1} \mathbf{E}[f(x_t) - f(x_*)] &\leq (L_0 - \mu)D_h(x_*, x_0) - c_{k-1}L_k \mathbf{E}[D_h(x_*, x_k)] + \sigma^2 \sum_{t=0}^{k-1} \frac{c_t}{L_t} \\
&\leq (L_0 - \mu)D_h(x_*, x_0) + \sigma^2 \sum_{t=0}^{k-1} \frac{c_t}{L_t}.
\end{aligned}$$

Dividing by C_k we get the desired result. \square

Theorem 5.4 itself does not provide insight about the convergence rate of Algorithm 4, as it strongly depends on the choice of stepsize parameters $\{L_t\}$. We study a suitable choice of stepsize rule in the next subsection.

5.5 Choice of stepsizes for Theorem 5.4

The goal of this section is to study a choice of stepsize parameters in Theorem 5.4. We will analyze separately two cases – $\mu = 0$ and $\mu > 0$.

Firstly we start with non-strongly convex case $\mu = 0$. The following lemma provides us with the choice of stepsizes minimizing right hand side of (29) - stepsizes giving us the best possible convergence rate for Theorem 5.4.

Corollary 5.5 (Nonstrongly convex rate for Algorithm 4). Suppose that $\mu = 0$, i.e. f is convex but not necessarily relative strongly convex. Suppose that we intend to run k iterations of Algorithm 4. Then, constant stepsize controlling parameters L_t given by

$$L_t = \frac{\sigma^2 L(k-1)}{-\sigma^2 + \sqrt{\sigma^4 + \sigma^2 AL(k-1)}} = \mathcal{O}(k^{-\frac{1}{2}})$$

minimize LHS of (29), obtaining

$$\sum_{t=1}^k \frac{\mathbf{E}[f(x_t) - f(x_*)]}{k} \leq \mathcal{O}(k^{-\frac{1}{2}}).$$

Note that Stochastic Gradient Descent in the standard smooth setting given by (25) with constant stepsize rule depending on the number of iterations enjoys $\mathcal{O}(1/\sqrt{k})$ rate as well [36].

Let us now proceed with the case $\mu > 0$. Note that the average of iterates of Stochastic Gradient Descent in the standard smooth setting given by (25) with stepsize $\gamma_t = \frac{1}{\mu t}$ enjoys $\mathcal{O}(\log(k)/k)$ rate [36]. Employing tail averaging technique one can obtain $\mathcal{O}(1/k)$ rate [29].

Lemma 5.6 (Choice of stepsizes for Algorithm 4). Suppose that sequence $\{L_t\}$ is nondecreasing and that sequence $\{c_t\}$ is monotonic for $t \geq T$. In order to attain $\mathcal{O}(1/\epsilon)$ rate for stochastic gradient descent we must have $L_t = \Theta(t)$.

Lemma 5.6 provides us with an insight on how stepsizes in Theorem 5.4 should be chosen in order to attain $\mathcal{O}(1/k)$ convergence rate - sequence of stepsize controlling parameters $\{L_t\}$ should be upper

and lower bounded by linear function in t . A faster or slower rate of increase of $\{L_t\}$ would not result in $O(1/k)$ convergence rate as $k \rightarrow \infty$.

The following lemma provides us a bound on convergence rate of Randomized Stochastic Gradient Descent, when sequence $\{L_t\}$ increases linearly with $L_0 = L$ i.e. $L_t = L + \alpha t$ for some $\alpha > 0$.

Lemma 5.7 (Linearly increasing stepsize parameters for Algorithm 4). Consider the convergence rate given by Theorem 5.4 and stepsize parameters given by $L_t = L + \alpha t$ for some $\alpha > 0$. Define

$$m_\mu \stackrel{\text{def}}{=} \max(\alpha, \mu - \alpha). \quad (30)$$

If we choose $\alpha > \mu$ then

$$\begin{aligned} C_k &\geq (L - \mu)^{1 - \frac{\mu}{\alpha}} \frac{(L - \mu + (k + 1)\alpha)^{\frac{\mu}{\alpha}} - (L - \mu + \alpha)^{\frac{\mu}{\alpha}}}{\mu}, \\ \sum_{t=0}^{k-1} \frac{c_t}{L_t} &\leq \frac{1}{L} + (L - \mu + \alpha)^{1 - \frac{\mu}{\alpha}} \frac{(L - \mu)^{\frac{\mu}{\alpha} - 1} - (L - \mu + k\alpha)^{\frac{\mu}{\alpha} - 1}}{\alpha - \mu}, \end{aligned}$$

if $\alpha = \mu$ then

$$\begin{aligned} C_k &= k, \\ \sum_{t=0}^{k-1} \frac{c_t}{L_t} &\leq \frac{\log(L + k\mu) - \log(L)}{\mu} + \frac{1}{L}, \end{aligned}$$

and finally if $\alpha < \mu$, then

$$\begin{aligned} C_k &\geq 1 + \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{(L - m_\mu + (k - 1)\alpha)^{\frac{\mu}{\alpha}} - (L - m_\mu)^{\frac{\mu}{\alpha}}}{\mu}, \\ \sum_{t=0}^{k-1} \frac{c_t}{L_t} &\leq \frac{1}{L} + \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{(L + k\alpha)^{\frac{\mu}{\alpha} - 1} - L^{\frac{\mu}{\alpha} - 1}}{\mu - \alpha}, \end{aligned}$$

for function Γ_α defined by (38). In the special case where $\alpha = \frac{\mu}{2}$ we obtain

$$\sum_{t=1}^k \frac{c_{t-1}}{C_k} \mathbf{E}[f(x_t) - f(x_*)] \leq \frac{(L - \mu)(L - \frac{\mu}{2})\mu D_h(x_*, x_0) + \sigma^2 \mu (1 - \frac{\mu}{2L} + k)}{(L + (k - 2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2 + (L - \frac{\mu}{2})\mu},$$

where

$$c_t = \frac{L + \frac{\mu}{2}(t - 1)}{L - \frac{\mu}{2}}, \quad C_k = \sum_{t=0}^{k-1} c_t.$$

Lemma 5.7 provides us with an useful insight on the linearly increasing choice of stepsize controlling parameters in Theorem 5.4. We consider the following 3 cases:

- $\alpha > \mu$: since $C_k = \Omega(k^{\mu/\alpha})$ and $\sum_{t=0}^{k-1} \frac{c_t}{L_t} = O(1)$, the convergence rate of the weighted sum of errors in functional values is $O(1/k^{\mu/\alpha})$. This is worse than the rate of stochastic gradient descent in standard smooth setting. However, weights from left hand side of the Theorem 5.4 are decreasing in this case.
- $\alpha = \mu$: Since $C_k = \Omega(k)$ and $\sum_{t=0}^{k-1} \frac{c_t}{L_t} = O(\log(k))$, the convergence rate of the weighted sum of errors in functional values is $O(\log(k)/k)$. Note that weighted sum of errors in objective from left hand side of (29) is an average in this case. The average of iterates of Stochastic Gradient Descent with stepsize parameters $L_t = \mu t$ under standard strong convexity assumption enjoys $O(\log(k)/k)$ rate as well [36].
- $\alpha < \mu$: Since $C_k = \Omega(k^{\mu/\alpha})$ and $\sum_{t=0}^{k-1} \frac{c_t}{L_t} = O(k^{\mu/\alpha-1})$ the convergence rate of the weighted sum of errors in functional values is $O(1/k)$. This is as good as the performance of Stochastic Gradient Descent in the standard smooth setting with tail averaging technique [29]. Note that weights from left hand side of the Theorem 5.4 are increasing, thus we put more value to latter iterates which has a similar effect to the convergence rate as tail averaging in the standard smooth setting. Recall that we use stepsize parameters given by $L_t = L + \alpha t$ for $\alpha < \mu$ in contrast of $L_t = \mu t$ used in [29] (rewritten to our notation).

The desired $O(1/k)$ convergence rate is obtained for $\alpha < \mu$. In practice, the condition $\alpha < \mu$ is not trivial to be satisfied, as the relative strong convexity parameter μ might be unknown and eventually very small. However, this issue can be overcome when strongly convex regularization is used - as we are aware of strongly convex parameter in this case.

5.6 Minibatch relSGD

As mentioned previously, if h is μ_h -strongly convex, σ^2 from the Assumption 5.1 can be chosen so that $\mu_h \sigma^2$ is a global upper bound on the variance of \tilde{g}_t .

Suppose that for $i = 1, 2, \dots, \tau$ random variables \tilde{g}_t^i are independent unbiased estimators of $\nabla f(x_t)$ coming from the same distribution.

Clearly, $\frac{1}{\tau} \sum_{i=1}^{\tau} \tilde{g}_t^i$ is an unbiased estimator of $\nabla f(x_t)$, thus we can set it in the update rule in Algorithm 4. Note that $\frac{1}{\tau} \sum_{i=1}^{\tau} \tilde{g}_t^i$ has τ times smaller variance comparing to \tilde{g}_t^i for all $i \leq \tau$. Thus, if we choose σ^2 such that $\sigma^2 \mu_h$ is an upper bound on the variance, we can allow it to be τ times smaller when using minibatch of size τ .

Corollary 5.8 (Convergence of Minibatch relSGD). Suppose that f is L smooth and μ strongly convex relative to μ_h strongly convex function h . Define $c_1 = 1$ and $c_t = \frac{L_{t-1}}{L_t - \mu} c_{t-1}$ for $t \geq 2$ and $C_k = \sum_{t=1}^k c_{t-1}$. Assume that variance unbiased gradient estimator \tilde{g}_t^i of $\nabla f(x_t)$ is upper bounded by $\sigma^2 \mu_h$ for all $i \leq \tau$ and $t \leq k$ and also that \tilde{g}_t^i are independent and identically distributed random variables. Then, iterates of Algorithm 4 with gradient estimator $\frac{1}{\tau} \sum_{i=1}^{\tau} \tilde{g}_t^i$ satisfy:

$$\sum_{t=1}^k \frac{c_{t-1}}{C_k} \mathbf{E}[f(x_t) - f(x_*)] \leq \frac{(L_0 - \mu) D_h(x_*, x_0)}{C_k} + \frac{\sigma^2}{\tau} \sum_{t=1}^{k-1} \frac{c_t}{C_k L_t}.$$

Let us consider a stepsize rule which yields $O(1/k)$ convergence rate as obtained from Lemma 5.7. In this case, τ -minibatching does not bring linear speedup in terms of the total number of iteration

to attain desired accuracy, and thus in terms of the actual work done by the algorithm, it is the best to choose smallest possible minibatch $\tau = 1$. However, minibatching can be particularly useful in the parallel setup - when one can obtain the a multiple gradient estimator by different processors at the same time.

6 Experiments

In this short section we numerically test the convergence of relGD, relRCD and relSGD on two artificial examples, in order to illustrate their potential.

6.1 An experiment with relRCD

In this example we compare standard gradient descent to relGD an relRCD. Recall that relRCD is always at most as fast relCD, once it can be applied. Our first experiment illustrates the need of relative smoothness assumption - as gradient descent with fixed stepsize applied on the considered function is extremely slow.

Let us consider a function

$$f(x) \stackrel{\text{def}}{=} \frac{1}{2}x^\top Mx + \frac{1}{10} \sum_{i=1}^{100} (x^{(i)})^4,$$

where $x \in \mathbb{R}^{100}$ and

$$M = \frac{A^\top A}{\lambda_{\max}(A^\top A)}.$$

Above, $A \in \mathbb{R}^{n \times n}$ is a random matrix with entries from normal distribution with zero mean and variance 1.

We will use the following reference function

$$h(x) \stackrel{\text{def}}{=} \frac{1}{2}\|x\|^2 + \frac{1}{10} \sum_{i=1}^{100} (x^{(i)})^4.$$

From Lemma 4.3 we know that f is 1-smooth relative to h . On top of that, $(f, \hat{S}) \sim \text{ESO}_h(v)$ with v such that $v^{(i)} = \max(\frac{1}{10}, (A^\top A)_{ii})$ and uniform sampling \hat{S} such that $\mathbf{P}(i \in \hat{S}) = 1/100$ for all i .

In order to compare relGD and relRCD to gradient descent, we need to find a (standard) smoothness parameter L . For this purpose, we will restrict the domain as $\{x \mid \|x\|_\infty^2 \leq 2\|x_0\|_\infty^2\}$. Clearly, $\frac{1}{2}x^\top Mx$ is 1-smooth and maximal eigenvalue of hessian of $\frac{1}{10} \sum_{i=1}^{100} (x^{(i)})^4$ is $\frac{12}{10}\|x\|_\infty^2$. We set x_0 to be random vector with independent zero mean entries with variance 10^6 . Thus, L is in the order of 10^6 in contrast to relative smoothness parameter, which is 1. The plot below illustrates a convergence result of gradient descent, relGD and relRCD for the artificial setting that we just described.

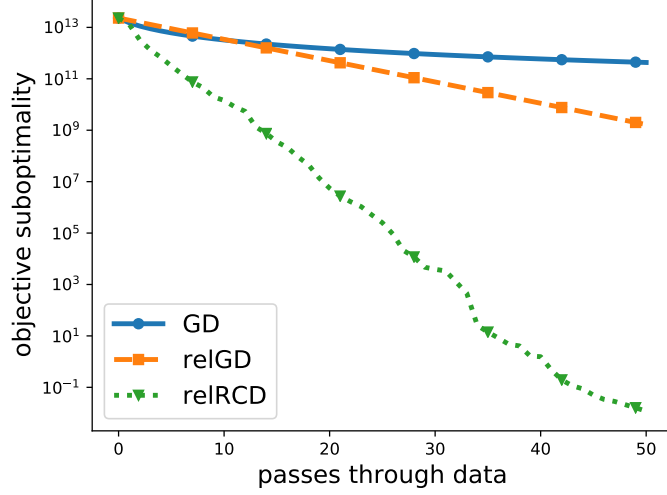


Figure 1: Comparison of Gradient descent to relGD and relRCD

Figure 1 show that the algorithms behave as we expected from the theory - Gradient descent has a faster start first few epochs, which is due to the fact that smoothness parameter L is huge but still tight in the region far from the optimum. However, with increasing number of iterations, Gradient descent is significantly outperformed by the other two algorithms. Notice that relRCD enjoys here the best convergence rate, which is expected from the theory since ESO parameters v are smaller than relative smoothness parameter. In this specific case maximal element of v is 0.36.

6.2 An experiment with relSGD

In this experiment we compare relGD to relSGD for various choice of stepsize parameters L_t .

Let us consider Poisson linear inverse problem, where one minimizes Kullback-Liebler divergence between b and Ax :

$$\begin{aligned} \min_x \quad & f(x) \stackrel{\text{def}}{=} \sum_{i=1}^m f^{(i)}(x) \stackrel{\text{def}}{=} \sum_{i=1}^m \left(b^{(i)} \log \frac{b^{(i)}}{(Ax)^{(i)}} + (Ax)^{(i)} - b^{(i)} \right) \\ \text{subject to} \quad & 0 < x^i, \forall i, \end{aligned}$$

where $b \in \mathbb{R}_{++}^m$ and matrix $A \in \mathbb{R}_+^{m \times n}$ have nonzero rows. In [3], it was shown that f is $L \stackrel{\text{def}}{=} \sum_{i=1}^m b^{(i)}$ -smooth with respect to Burg's entropy $h(x) \stackrel{\text{def}}{=} -\sum_{i=1}^m \log(x^{(i)})$.

We consider here $m \nabla f^{(i)}(x)$ for randomly chosen i to be an unbiased gradient estimator. Notice that the access to stochastic oracle is it is m times cheaper comparing to the cost of the full gradient due to ERM structure.

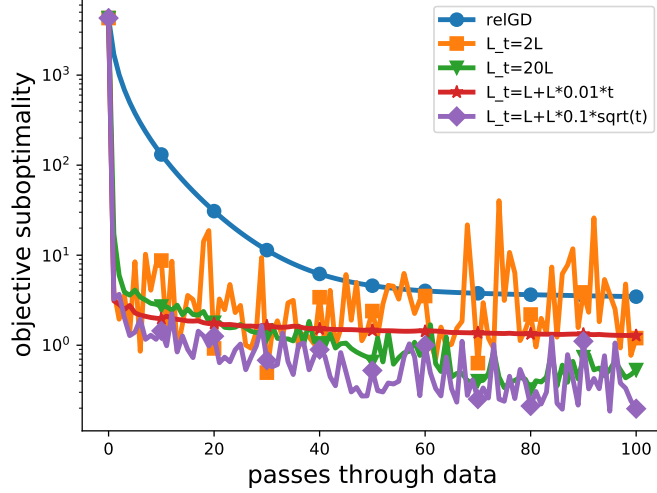


Figure 2: Comparison of relGD and relSGD for $A = |A'|$, $b = |b'|$, $x_0 = |x'_0|$ where A' , b' , x'_0 are vectors (or matrix) with entries randomly generated from normal distribution with zero mean and variance 1.

Figure 2 illustrates $O(1/k)$ convergence rate (sublinear) of relGD. We can clearly see that relSGD performs much faster first few passes through data, however for smaller constant L_t it oscillates and seems not converge to the optimum, as expected from theory. Larger constant L_t yields a very slightly slower decrease at the beginning but it is as expected less noisy and give us a better approximation after more passes through data. On the other hand, linearly increasing parameters L_t seems to be too fast, as the convergence significantly slows with the increased number of iterations. We obtained the best behaviour for $L_t = \frac{L}{10}\sqrt{t}$, which is expected since Corollary 5.5 claims that optimal stepsize controlling parameters are $O(1/\sqrt{k})$ for non-strongly convex case, i.e. $\mu = 0$.

7 Conclusions and Extensions

In this work, we presented first stochastic primal algorithms for minimizing Relatively smooth functions. We bridge the well developed area of stochastic smooth optimization with fresh area of relative smooth optimization. This way, we also contribute to better understanding of mirror descent, obtaining the first stochastic mirror descent type algorithm with linear convergence rate. However, there is still a plenty of space to extend on the results of our work. We give here few examples.

- Arbitrary Sampling for relRCD. In this work we showed the convergence of Randomized Coordinate Descent under ESO assumption for uniform sampling strategies. However, Randomized Coordinate Descent under standard smoothness allows arbitrary sampling strategy [30], which can potentially be extended to relative smooth setting as well, and therefore to gain additional speedup from importance sampling.
- Variance reduced relSGD for empirical risk minimization. RelSGD converges since the sequence of stepsize controlling parameters $\{L_t\}$ goes to infinity. However, for Empirical Risk Minimization problem in standard smooth setting, one can attain a linear convergence using variance reduction techniques [35, 8, 12, 37, 25], as we mentioned earlier.

- Application. In this work we provide only theoretical results on the algorithm and convergence rates. We did not give any application of our algorithms to a particular problem, however we believe that this work might help to solve a various optimization challenges in practice, especially since it brings a different insights on under which conditions can stochastic mirror descent perform extremely fast.

References

- [1] Arash Afkanpour, András György, Csaba Szepesvári, and Michael Bowling. A randomized mirror descent algorithm for large scale multiple kernel learning. In *International Conference on Machine Learning*, pages 374–382, 2013.
- [2] Zeyuan Allen-Zhu and Lorenzo Orecchia. Linear coupling: An ultimate unification of gradient and mirror descent. *arXiv preprint arXiv:1407.1537*, 2014.
- [3] Heinz H. Bauschke, Jérôme Bolte, and Marc Teboulle. A descent lemma beyond Lipschitz gradient continuity: first-order methods revisited and applications. *Mathematics of Operations Research*, pages 330–348, 2016.
- [4] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
- [5] Martin Benning, Marta Betcke, Matthias Ehrhardt, and Carola-Bibiane Schönlieb. Gradient descent in a generalised Bregman distance framework. *arXiv preprint arXiv:1612.02506*, 2016.
- [6] Benjamin Birnbaum, Nikhil R Devanur, and Lin Xiao. Distributed algorithms via gradient descent for Fisher markets. In *Proceedings of the 12th ACM conference on Electronic commerce*, pages 127–136. ACM, 2011.
- [7] Cong D Dang and Guanghui Lan. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881, 2015.
- [8] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. *arXiv:1407.0202*, 2014.
- [9] Nicolas Flammarion and Francis Bach. Stochastic composite least-squares regression with convergence rate $\mathcal{O}(1/n)$. *arXiv preprint arXiv:1702.06429*, 2017.
- [10] Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22(4):1469–1492, 2012.
- [11] Le Thi Khanh Hien, Canyi Lu, Huan Xu, and Jiashi Feng. Accelerated stochastic mirror descent algorithms for composite non-strongly convex optimization. *arXiv preprint arXiv:1605.06892*, 2016.
- [12] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.
- [13] Lange Kenneth. *MM optimization algorithms*. SIAM, 2016.

- [14] Diederik P. Kingma and Jimmy Ba. Adam: a method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.
- [15] Walid Krichene, Alexandre Bayen, and Peter L Bartlett. Accelerated mirror descent in continuous and discrete time. In *Advances in neural information processing systems*, pages 2845–2853, 2015.
- [16] Guanghui Lan, Zhaosong Lu, and Renato D. C. Monteiro. Primal-dual first-order methods with $\mathcal{O}(1/\epsilon)$ iteration-complexity for cone programming. *Mathematical Programming*, 126(1):1–29, 2011.
- [17] Haihao Lu. “Relative-continuity” for non-Lipschitz non-smooth convex optimization using stochastic (or deterministic) mirror descent. *arXiv preprint arXiv:1710.04718*, 2017.
- [18] Haihao Lu, Robert M. Freund, and Yurii Nesterov. Relatively-smooth convex optimization by first-order methods, and applications. *arXiv preprint arXiv:1610.05708*, 2016.
- [19] Angelia Nedic and Soomin Lee. On stochastic subgradient mirror-descent algorithm with weighted averaging. *SIAM Journal on Optimization*, 24(1):84–107, 2014.
- [20] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.
- [21] Arkadi Nemirovsky and David B. Yudin. *Problem complexity and method efficiency in optimization*. Wiley, New York, 1983.
- [22] Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.
- [23] Yurii Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.
- [24] Yurii Nesterov. *Introductory lectures on convex optimization: a basic course*. Kluwer Academic Publishers, 2004.
- [25] Lam Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. *arXiv preprint arXiv:1703.00102*, 2017.
- [26] Boris T Polyak. *Introduction to Optimization*. Optimization Software, 1987.
- [27] Zheng Qu and Peter Richtárik. Coordinate descent with arbitrary sampling I: Algorithms and complexity. *Optimization Methods and Software*, 31(5):829–857, 2016.
- [28] Zheng Qu and Peter Richtárik. Coordinate descent with arbitrary sampling II: Expected separable overapproximation. *Optimization Methods and Software*, 31(5):858–884, 2016.
- [29] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *Proceedings of the 29th International Conference on Machine Learning*, pages 449–456, 2012.
- [30] Peter Richtárik and Martin Takáč. On optimal probabilities in stochastic coordinate descent methods. *Optimization Letters*, 10(6):1233–1243, 2016.

- [31] Peter Richtárik and Martin Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1-2):433–484, 2016.
- [32] Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144:1–38, 2014.
- [33] Peter Richtárik and Martin Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, 156(1):433–484, 2016.
- [34] H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.
- [35] Nicolas Le Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems*, pages 2663–2671, 2012.
- [36] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: from theory to algorithms*. Cambridge University Press, 2014.
- [37] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss. *J. Mach. Learn. Res.*, 14(1):567–599, February 2013.
- [38] Rachael Tappenden, Martin Takáč, and Peter Richtárik. On the complexity of parallel coordinate descent. *arXiv preprint arXiv:1503.03033*, 2015.
- [39] Paul Tseng. On accelerated proximal gradient methods for convex-concave optimization. *Submitted to SIAM Journal on Optimization*, 2008.
- [40] Li Zhang. Proportional response dynamics in the Fisher market. *Theoretical Computer Science*, 412(24):2691 – 2698, 2011. Selected Papers from 36th International Colloquium on Automata, Languages and Programming (ICALP 2009).

A Key technical lemmas

For completeness, we firstly give proof of Three point property.

A.1 Proof of the three point property

Note that $\phi(x) + D_h(x, z)$ is differentiable and convex in x . Using the definition of z_+ we have

$$\langle \nabla \phi(z_+) + \nabla h(z_+) - \nabla h(z), x - z_+ \rangle \geq 0, \quad \forall x \in Q.$$

Using definition of $D_h(\cdot, \cdot)$ we can see that

$$\langle \nabla h(z_+) - \nabla h(z), x - z_+ \rangle = D_h(x, z) - D_h(z_+, z) - D_h(x, z_+).$$

Putting the above together, we see that

$$\begin{aligned} 0 &\leq \langle \nabla \phi(z_+) + \nabla h(z_+) - \nabla h(z), x - z_+ \rangle \\ &= D_h(x, z) - D_h(z_+, z) - D_h(x, z_+) + \langle \nabla \phi(z_+), x - z_+ \rangle \\ &\leq D_h(x, z) - D_h(z_+, z) - D_h(x, z_+) + \phi(x) - \phi(z_+). \end{aligned}$$

The last inequality is due to convexity of ϕ .

A.2 Key lemma for analysis

The following lemma allow us to get a convergence rate for Algorithms

Lemma A.1. Suppose that for positive sequences $\{f_t\}, \{D_t\}$ we have

$$f_{t+1} \leq (1 - \delta)f_t + \delta f_* + (\varphi - \delta\psi) D_t - \varphi D_{t+1}, \quad (31)$$

where $\delta, \varphi, \psi \in \mathbb{R}$ satisfy $1 \geq \delta > 0$ and $\varphi \geq \psi > 0$. Then, the following inequality holds

$$\sum_{t=1}^k c_t (f_t - f_*) \leq \frac{(\varphi - \delta\psi)D_0 + (1 - \delta)(f_0 - f_*)}{1 - \frac{\varphi}{\psi} + \frac{\varphi}{\psi} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{k-1}},$$

where $c_t \stackrel{\text{def}}{=} C_t / \sum_{t=1}^k C_t$ for

$$C_t \stackrel{\text{def}}{=} \begin{cases} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{t-1} \frac{\varphi - \psi}{\delta - 1 - \varphi - \psi}, & 1 \leq t \leq k - 1 \\ \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{k-1}, & t = k. \end{cases}$$

Proof. Let us multiple the inequality (31) by $\left(\frac{\varphi}{\varphi - \delta\psi}\right)^t$ for iterates $t = 0, 1, \dots, k - 1$ and sum them:

$$\begin{aligned} \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^t f_{t+1} &\leq \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^t \left((1 - \delta)f_t + \delta f_* \right) \\ &\quad + \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^t \left((\varphi - \delta\psi) D_t - \varphi D_{t+1} \right). \end{aligned}$$

Rearranging the terms, we get

$$\sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^t (f_{t+1} - (1 - \delta)f_t - \delta f_*) \quad (32)$$

$$\begin{aligned} &\leq (\varphi - \delta\psi) D_0 - \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{k-1} \varphi D_k \\ &\leq (\varphi - \delta\psi) D_0. \end{aligned} \quad (33)$$

For simplicity, throughout this proof denote $r_t = f_t - f_*$. Let us continue with the bound above:

$$\begin{aligned} (\varphi - \delta\psi) D_0 &\stackrel{(33)}{\geq} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{k-1} f_k + \sum_{t=1}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{t-1} \left(f_t - (1 - \delta) \frac{\varphi}{\varphi - \delta\psi} f_t \right) \\ &\quad - (1 - \delta) f_0 - \delta \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^t f_* \\ &= \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{k-1} f_k + \sum_{t=1}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{t-1} \frac{\varphi - \psi}{\delta^{-1}\varphi - \psi} f_t \\ &\quad - (1 - \delta) f_0 - \delta \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^t f_* \end{aligned} \quad (34)$$

$$\stackrel{(*)}{=} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{k-1} r_k + \sum_{t=1}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{t-1} \frac{\varphi - \psi}{\delta^{-1}\varphi - \psi} r_t - (1 - \delta) r_0. \quad (35)$$

Equality (*) is obtained by the fact that the sum of terms corresponding to $f(\cdot)$ is 0 (this can be easily seen as it is equal to (32)).

Recall that we have

$$C_t = \begin{cases} \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{t-1} \frac{\varphi - \psi}{\delta^{-1}\varphi - \psi}, & 1 \leq t \leq k-1 \\ \left(\frac{\varphi}{\varphi - \delta\psi} \right)^{k-1}, & t = k. \end{cases}$$

and $c_t \stackrel{\text{def}}{=} C_t / \sum_{t=1}^k C_t$. Since the sum of terms corresponding to f_t for some t or f_* in (34) is 0 (because it is equal to (32)), we have

$$\begin{aligned}
\sum_{t=1}^k C_t &= \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{k-1} + \sum_{t=1}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{t-1} \frac{\varphi - \psi}{\frac{\eta}{\tau}\varphi - \psi} \\
&= (1 - \delta) + \delta \sum_{t=0}^{k-1} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^t. \\
&= (1 - \delta) + \delta \frac{\left(\frac{\varphi}{\varphi - \delta\psi}\right)^k - 1}{\frac{\varphi}{\varphi - \delta\psi} - 1} \\
&= (1 - \delta) + \frac{\left(\frac{\varphi}{\varphi - \delta\psi}\right)^k - 1}{\frac{\psi}{\varphi - \delta\psi}} \\
&= (1 - \delta) + (\varphi - \delta\psi) \frac{\left(\frac{\varphi}{\varphi - \delta\psi}\right)^k - 1}{\psi} \\
&= 1 - \frac{\varphi}{\psi} + \frac{\varphi}{\psi} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{k-1}. \tag{36}
\end{aligned}$$

Thus, we can rewrite (35) as follows

$$\begin{aligned}
\sum_{t=1}^k c_t r_t &\stackrel{(35)}{\leq} \left((\varphi - \delta\psi) D_0 + (1 - \delta)r_0 \right) \frac{1}{\sum_{t=1}^k C_t} \\
&\stackrel{(36)}{=} \left((\varphi - \delta\psi) D_0 + (1 - \delta)r_0 \right) \frac{1}{1 - \frac{\varphi}{\psi} + \frac{\varphi}{\psi} \left(\frac{\varphi}{\varphi - \delta\psi}\right)^{k-1}}.
\end{aligned}$$

□

B Proofs for Section 5

B.1 Proof of Corollary 5.5

Denote $l_t = (L_t)^{-1}$ for simplicity. It is easy to see that

$$c_t = L l_t, \quad C_k = 1 + L \sum_{t=1}^{k-1} l_t, \quad \sum_{t=0}^{k-1} c_t l_t = L + L \left(\sum_{t=1}^{k-1} l_t^2 \right).$$

Denote

$$A = (L - \mu)D_h(x_*, x_0) + \sigma^2 L.$$

Minimizing RHS of (29) to obtain the best rate is equivalent to minimize

$$\frac{A + \sigma^2 L \left(\sum_{t=1}^{k-1} l_t^2 \right)}{1 + L \sum_{t=1}^{k-1} l_t}.$$

Notice that the expression above is minimized for constant l_t , as if $l_t \neq l_s$, setting $l_t = l_s = \frac{l_t + l_s}{2}$ leads to strictly smaller value of the expression. Therefore, it suffices to minimize

$$\frac{A + \sigma^2 L(k-1)l^2}{1 + L(k-1)l}$$

in l . First order optimality condition yields

$$2\sigma^2 L(k-1)l(1 + L(k-1)l) = (A + \sigma^2 L(k-1)l^2)L(k-1),$$

which is equivalent to

$$\sigma^2 L(k-1)l^2 + 2\sigma^2 l - A = 0.$$

The quadratic equation above have a single solution

$$l = \frac{-\sigma^2 + \sqrt{\sigma^4 + \sigma^2 AL(k-1)}}{\sigma^2 L(k-1)},$$

which finishes the proof.

B.2 Proof of Lemma 5.6

For simplicity, denote $l_t = (L_t^{-1})$. Thus, $\{l_t\}$ is nonincreasing sequence. Note that the rate from the Theorem 5.4 is $O(1/k)$ if and only if both

$$\frac{1}{C_k} \quad \text{and} \quad \sum_{t=0}^{k-1} \frac{c_t l_t}{C_k}$$

are $O(1/k)$.

Let us now consider that $\{c_t\}$ is nonincreasing for $t \geq T$. Suppose that

$$1 > \liminf \frac{c_t}{c_{t-1}} \stackrel{\text{def}}{=} r_c.$$

Then for all k there is $K \geq k$ such that

$$1 > \frac{1 + r_c}{2} > \frac{c_K}{c_{K-1}}.$$

Thus there is infinitely many t such that

$$1 > \frac{1 + r_c}{2} > \frac{c_t}{c_{t-1}}.$$

Since $\{c_t\}$ is nonincreasing for $t \geq T$, we have that $\{c_t\} \rightarrow 0$ which is a contradiction with the assumption that $\frac{1}{C_t} = O(1/t)$. Thus we have

$$1 = \liminf \frac{c_t}{c_{t-1}} = \lim \frac{c_t}{c_{t-1}},$$

which implies that

$$\lim L_t - L_{t-1} = \mu.$$

The above means that $L_t = \Theta(t)$. We have just proven the lemma for asymptotically nonincreasing $\{c_t\}$.

Now, suppose that $\{c_t\}$ is increasing sequence for $t \geq T$. Then we have for all $t \geq T$

$$\frac{L_{t-1}}{L_t - \mu} > 1.$$

Thus $L_t < L_{t-1} + \mu$, which implies that $L_t = O(t)$ and $l_t = \Omega(1/t)$.

On the other hand, looking at $\sum_{t=0}^{k-1} \frac{c_t l_t}{C_k}$ as the weighted sum of l_t , since l_{k-1} is the smallest from $\{l_t\}$ we immediately have

$$O(1/k) = \sum_{t=0}^{k-1} \frac{c_t l_t}{C_k} \geq l_{k-1} \geq l_k,$$

which means that $l_t = O(1/t)$. Thus, $l_t = \Theta(1/t)$ and $L_t = \Theta(t)$.

B.3 Proof of Lemma 5.7

First, we introduce two technical lemmas.

Lemma B.1. Let us fix $\alpha > 0$. There exist a convex continuous function $\gamma_\alpha(x)$ on \mathbb{R}_+ such that for all $x > 0$ we have

$$\gamma_\alpha(x + \alpha) = \log(x) + \gamma_\alpha(x). \quad (37)$$

Proof. We will construct function γ_α in the following way - Let us set $\gamma_\alpha(x) = 0$ for $x \in [1, 1 + \alpha)$. For $x \geq 1 + \alpha$ let us set recursively $\gamma_\alpha(x + \alpha) = \log(x) + \gamma_\alpha(x)$ and for $x < 1$ let us set $\gamma_\alpha(x) = -\log(x)$. Clearly, equality (37) holds.

We will firstly prove that γ_α is continuous on \mathbb{R}_+ and differentiable on $\mathbb{R}_+ \setminus \{1\}$. Let us start with intervals $[1 + k\alpha, 1 + (k + 1)\alpha)$ for all k .

Clearly, γ_α it is continuous and differentiable on $[1, 1 + \alpha)$. Suppose now inductively that γ_α is continuous and differentiable on $[1 + k\alpha, 1 + (k + 1)\alpha)$ for some $k \geq 0$. Then, for $x \in [1 + (k + 1)\alpha, 1 + (k + 2)\alpha)$ we have

$$\gamma_\alpha(x) = \log(x - \alpha) + \gamma_\alpha(x - \alpha).$$

Since both $\log(x - \alpha)$ and $\gamma_\alpha(x - \alpha)$ are continuous and differentiable functions on $[1 + (k + 1)\alpha, 1 + (k + 2)\alpha)$, $\gamma_\alpha(x)$ is also continuous and differentiable on $[1 + (k + 1)\alpha, 1 + (k + 2)\alpha)$.

Clearly, γ_α it is continuous and differentiable on $(0, 1)$.

It remains to show continuity and differentiability in the points $\{1 + k\alpha\}$ for $k \geq 1$ and continuity in $\{1\}$. It is a simple exercise to see the continuity and differentiability in $\{1 + \alpha\}$. For $1 + k\alpha$ where $k \geq 2$ we can show it inductively - as $\gamma_\alpha(x - \alpha)$ and $\log(x - \alpha)$ are continuous and differentiable on $(1 + (k - \frac{1}{2})\alpha, 1 + (k + \frac{1}{2})\alpha)$, then $\gamma_\alpha(x)$ is continuous and differentiable on $(1 + (k - \frac{1}{2})\alpha, 1 + (k + \frac{1}{2})\alpha)$ as well and thus it is continuous and differentiable in point $\{1 + k\alpha\}$. On top of that, γ_α is clearly continuous in $\{1\}$.

We have just proven that γ_α is continuous on \mathbb{R}_+ and differentiable on $\mathbb{R}_+ \setminus \{1\}$.

Now we can proceed with the proof of convexity. We will show that the (sub)derivative of γ_α is nonnegative for all $x > 0$. Clearly, $\gamma'_\alpha(x) \geq 0$ for $x \in (0, 1)$ and subdifferential in $\{1\}$ is nonnegative as well. Let us write $x = 1 + \{x\}_\alpha + k\alpha$, where $0 \leq \{x\}_\alpha < \alpha$ and $k \geq -1$. Then we have

$$\begin{aligned}
\gamma'_\alpha(x) &= \lim_{\epsilon \rightarrow 0} \frac{\gamma_\alpha(x + \epsilon) - \gamma_\alpha(x)}{\epsilon} \\
&= \lim_{\epsilon \rightarrow 0} \frac{\sum_{i=0}^{k-1} (\log(1 + \{x\}_\alpha + i\alpha + \epsilon) - \log(1 + \{x\}_\alpha + i\alpha))}{\epsilon} \\
&\quad + \frac{\gamma_\alpha(1 + \{x\}_\alpha + \epsilon) - \gamma_\alpha(1 + \{x\}_\alpha)}{\epsilon} \\
&\stackrel{(*)}{=} \lim_{\epsilon \rightarrow 0} \frac{\sum_{i=0}^{k-1} (\log(1 + \{x\}_\alpha + i\alpha + \epsilon) - \log(1 + \{x\}_\alpha + i\alpha))}{\epsilon} \\
&\stackrel{(**)}{\geq} 0.
\end{aligned}$$

Equality (*) holds since for small enough ϵ we have $1 + \{x\}_\alpha + \epsilon < 2\alpha$ and inequality (**) holds due to the fact that logarithm is an increasing function. \square

Denote

$$\Gamma_\alpha(x) \stackrel{\text{def}}{=} \exp(\gamma_\alpha(x)) \quad (38)$$

for γ_α given from Lemma B.1. Thus, Γ_α is log-convex function satisfying

$$\Gamma_\alpha(x + \alpha) = x\Gamma_\alpha(x). \quad (39)$$

Note that when $\alpha = 1$, function γ can be chosen as log Gamma function and thus Γ_1 can be chosen to be standard Gamma function.

The following lemma is crucial for our analysis, allowing us to bound the ratio of functions $\Gamma_\alpha(\cdot)$ with nearby arguments.

Lemma B.2. Consider a function Γ_α defined above. Then, we have for all $0 \leq s \leq \alpha$ and $x > 0$:

$$x^{1-\frac{s}{\alpha}} \leq \frac{\Gamma_\alpha(x + \alpha)}{\Gamma_\alpha(x + s)} \leq (x + \alpha)^{1-\frac{s}{\alpha}}. \quad (40)$$

Proof. Using convexity of γ_α we have

$$\Gamma_\alpha(x + s) \leq \Gamma_\alpha(x)^{1-\frac{s}{\alpha}} \Gamma_\alpha(x + \alpha)^{\frac{s}{\alpha}} \stackrel{(39)}{=} x^{\frac{s}{\alpha}-1} \Gamma_\alpha(x + \alpha).$$

Rearranging the above we obtain

$$x^{1-\frac{s}{\alpha}} \leq \frac{\Gamma_\alpha(x + \alpha)}{\Gamma_\alpha(x + s)}.$$

On the other hand, using convexity of γ_α again we obtain

$$\Gamma_\alpha(x + \alpha) \leq \Gamma_\alpha(x + s)^{\frac{s}{\alpha}} \Gamma_\alpha(x + s + \alpha)^{1-\frac{s}{\alpha}} \stackrel{39}{=} (x + s)^{1-\frac{s}{\alpha}} \Gamma_\alpha(x + s).$$

By rearranging the above, we get

$$\frac{\Gamma_\alpha(x + \alpha)}{\Gamma_\alpha(x + s)} \leq (x + s)^{1-\frac{s}{\alpha}} \leq (x + \alpha)^{1-\frac{s}{\alpha}}.$$

\square

We can now proceed with the proof of Lemma 5.7 itself.

Proof. Note that

$$\begin{aligned}
c_t &= \prod_{i=0}^{t-1} \frac{L_i}{L_{i+1} - \mu} \\
&\stackrel{(39)}{=} \frac{\frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L)}}{\frac{\Gamma_\alpha(L+(t+1)\alpha-\mu)}{\Gamma_\alpha(L-\mu+\alpha)}} \\
&= \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\mu+(t+1)\alpha)}. \tag{41}
\end{aligned}$$

Let us firstly consider the case when $\alpha > \mu$. Choosing $x = L - \mu + t\alpha$ and $s = \mu$ in (40) we get

$$(L - \mu + t\alpha)^{1-\frac{\mu}{\alpha}} \leq \frac{\Gamma_\alpha(L - \mu + (t+1)\alpha)}{\Gamma_\alpha(L + t\alpha)} \leq (L - \mu + (t+1)\alpha)^{1-\frac{\mu}{\alpha}}.$$

The inequality above allows us to get the following bound on c_t

$$\frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L - \mu + t\alpha)^{\frac{\mu}{\alpha}-1} \geq c_t \geq \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L - \mu + (t+1)\alpha)^{\frac{\mu}{\alpha}-1}. \tag{42}$$

Clearly, $\{c_t\}$ is decreasing and thus using the bound above we obtain

$$\begin{aligned}
C_k &= \sum_{t=0}^{k-1} c_t \stackrel{(42)}{\geq} \sum_{t=0}^{k-1} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L - \mu + (t+1)\alpha)^{\frac{\mu}{\alpha}-1} \\
&= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \sum_{t=0}^{k-1} (L - \mu + (t+1)\alpha)^{\frac{\mu}{\alpha}-1} \\
&\stackrel{(*)}{\geq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^k (L - \mu + (t+1)\alpha)^{\frac{\mu}{\alpha}-1} dt \\
&= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^{(k)\alpha} (L - \mu + \alpha + t)^{\frac{\mu}{\alpha}-1} \frac{1}{\alpha} dt \\
&= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{1}{\alpha} \left[\frac{(L - \mu + \alpha + t)^{\frac{\mu}{\alpha}}}{\frac{\mu}{\alpha}} \right]_{t=0}^{k\alpha} \\
&= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{(L - \mu + (k+1)\alpha)^{\frac{\mu}{\alpha}} - (L - \mu + \alpha)^{\frac{\mu}{\alpha}}}{\mu} \\
&\stackrel{(40)}{\geq} (L - \mu)^{1-\frac{\mu}{\alpha}} \frac{(L - \mu + (k+1)\alpha)^{\frac{\mu}{\alpha}} - (L - \mu + \alpha)^{\frac{\mu}{\alpha}}}{\mu}.
\end{aligned}$$

Inequality (*) holds since $(L - \mu + (t+1)\alpha)^{\mu/\alpha-1}$ is decreasing in t . On the other hand, we have

$$\begin{aligned}
\sum_{t=1}^{k-1} \frac{c_t}{L_t} &\stackrel{(42)}{\leq} \sum_{t=1}^{k-1} \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} (L-\mu+t\alpha)^{\frac{\mu}{\alpha}-1} \frac{1}{L+t\alpha} \\
&= \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \sum_{t=1}^{k-1} \frac{1}{L+t\alpha} (L-\mu+t\alpha)^{\frac{\mu}{\alpha}-1} \\
&\stackrel{(*)}{\leq} \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \sum_{t=1}^{k-1} (L-\mu+t\alpha)^{\frac{\mu}{\alpha}-2} \\
&\stackrel{(**)}{\leq} \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \int_0^k (L-\mu+t\alpha)^{\frac{\mu}{\alpha}-2} dt \\
&= \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \int_0^{k\alpha} (L-\mu+t)^{\frac{\mu}{\alpha}-2} \frac{1}{\alpha} dt \\
&= \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \frac{1}{\alpha} \left[\frac{(L-\mu+t)^{\frac{\mu}{\alpha}-1}}{\frac{\mu}{\alpha}-1} \right]_0^{k\alpha} \\
&= \frac{\Gamma_\alpha(L-\mu+\alpha)}{\Gamma_\alpha(L)} \frac{(L-\mu)^{\frac{\mu}{\alpha}-1} - (L-\mu+k\alpha)^{\frac{\mu}{\alpha}-1}}{\alpha-\mu} \\
&\stackrel{(40)}{\leq} (L-\mu+\alpha)^{1-\frac{\mu}{\alpha}} \frac{(L-\mu)^{\frac{\mu}{\alpha}-1} - (L-\mu+k\alpha)^{\frac{\mu}{\alpha}-1}}{\alpha-\mu}.
\end{aligned}$$

Inequality (*) holds due to the fact that $(L+t\alpha)^{-1} \leq (L-\mu+t\alpha)^{-1}$ and inequality (**) holds since $(L-\mu+t\alpha)^{\mu/\alpha-2}$ is decreasing in t . Thus we have

$$\sum_{t=0}^{k-1} \frac{c_t}{L_t} \leq \frac{1}{L} + (L-\mu+\alpha)^{1-\frac{\mu}{\alpha}} \frac{(L-\mu)^{\frac{\mu}{\alpha}-1} - (L-\mu+k\alpha)^{\frac{\mu}{\alpha}-1}}{\alpha-\mu}.$$

and we have just proven the first part of the lemma.

Let us now look at the case when $\alpha \leq \mu$. It will be useful to denote $\lfloor \mu \rfloor_\alpha$ as the largest integer such that $\mu - \lfloor \mu \rfloor_\alpha \alpha$ is positive. Denote also

$$\{\mu\}_\alpha \stackrel{\text{def}}{=} \mu - \lfloor \mu \rfloor_\alpha \alpha.$$

Using (39) we obtain

$$\begin{aligned}
\frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\mu+(t+1)\alpha)} &= \frac{\Gamma_\alpha(L+t\alpha)(L+t\alpha+\alpha-\mu)(L+t\alpha+2\alpha-\mu)\dots(L+t\alpha+(\lfloor \mu \rfloor_\alpha-1)\alpha-\mu)}{\Gamma_\alpha(L+t\alpha+\lfloor \mu \rfloor_\alpha\alpha-\mu)} \\
&= \frac{\Gamma_\alpha(L+t\alpha)(L+t\alpha+\alpha-\mu)(L+t\alpha+2\alpha-\mu)\dots(L-\{\mu\}_\alpha+(t-1)\alpha)}{\Gamma_\alpha(L-\{\mu\}_\alpha+t\alpha)}.
\end{aligned}$$

Upper and lower bounding the equality above we get

$$\frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\mu+(t+1)\alpha)} \geq \frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\{\mu\}_\alpha+t\alpha)} (L-\mu+(t+1)\alpha)^{\lfloor \mu \rfloor_\alpha-1}, \quad (43)$$

$$\frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\mu+(t+1)\alpha)} \leq \frac{\Gamma_\alpha(L+t\alpha)}{\Gamma_\alpha(L-\{\mu\}_\alpha+t\alpha)} (L-\{\mu\}_\alpha+(t-1)\alpha)^{\lfloor \mu \rfloor_\alpha-1}. \quad (44)$$

Using (40) we have

$$(L + (t-1)\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} \leq \frac{\Gamma_\alpha(L + t\alpha)}{\Gamma_\alpha(L - \{\mu\}_\alpha + t\alpha)} \leq (L + t\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}}. \quad (45)$$

Now we are ready to get upper and lower bound on c_t :

$$\begin{aligned} c_t &\stackrel{(41)}{=} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{\Gamma_\alpha(L + t\alpha)}{\Gamma_\alpha(L - \mu + (t+1)\alpha)} \\ &\stackrel{(43)}{\geq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{\Gamma_\alpha(L + t\alpha)}{\Gamma_\alpha(L - \{\mu\}_\alpha + t\alpha)} (L - \mu + (t+1)\alpha)^{|\mu|_\alpha - 1} \\ &\stackrel{(45)}{\geq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + (t-1)\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} (L - \mu + (t+1)\alpha)^{|\mu|_\alpha - 1}. \end{aligned} \quad (46)$$

$$\begin{aligned} c_t &\stackrel{(41)}{=} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{\Gamma_\alpha(L + t\alpha)}{\Gamma_\alpha(L - \mu + (t+1)\alpha)} \\ &\stackrel{(44)}{\leq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{\Gamma_\alpha(L + t\alpha)}{\Gamma_\alpha(L - \{\mu\}_\alpha + t\alpha)} (L - \{\mu\}_\alpha + (t-1)\alpha)^{|\mu|_\alpha - 1} \\ &\stackrel{(45)}{\leq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + t\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} (L - \{\mu\}_\alpha + (t-1)\alpha)^{|\mu|_\alpha - 1} \end{aligned} \quad (47)$$

Recall that we have $m_\mu = \max(\alpha, \mu - \alpha)$. Then, we can get the following bound on C_k :

$$\begin{aligned} C_k - c_0 &= \sum_{t=1}^{k-1} c_t \\ &\stackrel{(46)}{\geq} \sum_{t=1}^{k-1} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + (t-1)\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} (L - \mu + (t+1)\alpha)^{|\mu|_\alpha - 1} \\ &\stackrel{(30)}{\geq} \sum_{t=1}^{k-1} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L - m_\mu + t\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} (L - m_\mu + t\alpha)^{|\mu|_\alpha - 1} \\ &= \sum_{t=1}^{k-1} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L - m_\mu + t\alpha)^{\frac{\mu}{\alpha} - 1} \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \sum_{t=1}^{k-1} (L - m_\mu + t\alpha)^{\frac{\mu}{\alpha} - 1} \\ &\stackrel{(*)}{\geq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^{k-1} (L - m_\mu + t\alpha)^{\frac{\mu}{\alpha} - 1} dt \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^{(k-1)\alpha} (L - m_\mu + t)^{\frac{\mu}{\alpha} - 1} \frac{1}{\alpha} dt \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{1}{\alpha} \left[\frac{(L - m_\mu + t)^{\frac{\mu}{\alpha}}}{\frac{\mu}{\alpha}} \right]_{t=0}^{(k-1)\alpha} \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{(L - m_\mu + (k-1)\alpha)^{\frac{\mu}{\alpha}} - (L - m_\mu)^{\frac{\mu}{\alpha}}}{\mu}. \end{aligned} \quad (48)$$

Inequality (*) holds since $(L - m_\mu + t\alpha)^{\mu/\alpha-1}$ is increasing function. Note that in the case when $\alpha = \mu$, all bounds above hold with equality and we have

$$C_k = k.$$

To finish the proof of the second and third part of the Lemma, it remains to upper bound $\sum_{t=0}^{k-1} c_t L_t^{-1}$. Firstly, note that

$$\begin{aligned} \frac{c_t}{L_t} &\stackrel{(47)}{\leq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + t\alpha)^{\frac{\{\mu\}_\alpha}{\alpha}} (L - \{\mu\}_\alpha + (t-1)\alpha)^{|\mu|_\alpha-1} (L + t\alpha)^{-1} \\ &\stackrel{(*)}{\leq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + t\alpha)^{\frac{\mu}{\alpha}-1} (L + t\alpha)^{-1} \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + t\alpha)^{\frac{\mu}{\alpha}-2}. \end{aligned} \quad (49)$$

Inequality (*) holds due to the fact that $L - \{\mu\}_\alpha + (t-1)\alpha \leq L + t\alpha$. We can continue bounding as follows

$$\begin{aligned} \sum_{t=1}^{k-1} \frac{c_t}{L_t} &\stackrel{(49)}{\leq} \sum_{t=1}^{k-1} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} (L + t\alpha)^{\frac{\mu}{\alpha}-2} \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \sum_{t=1}^{k-1} (L + t\alpha)^{\frac{\mu}{\alpha}-2} \\ &\stackrel{(*)}{\leq} \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^k (L + t\alpha)^{\frac{\mu}{\alpha}-2} dt \\ &= \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \int_0^{k\alpha} (L + t)^{\frac{\mu}{\alpha}-2} \frac{1}{\alpha} dt \\ &\stackrel{(**)}{=} \begin{cases} \frac{\log(L+k\mu) - \log(L)}{\mu} & \text{if } \alpha = \mu, \\ \frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} \frac{(L+k\alpha)^{\frac{\mu}{\alpha}-1} - L^{\frac{\mu}{\alpha}-1}}{\mu - \alpha} & \text{if } \alpha < \mu. \end{cases} \end{aligned} \quad (50)$$

Inequality (*) holds due to the fact that for $\mu \geq 2\alpha$ we have

$$\sum_{t=1}^{k-1} (L + t\alpha)^{\frac{\mu}{\alpha}-2} \leq \int_1^k (L + t\alpha)^{\frac{\mu}{\alpha}-2} dt$$

and for $\mu < 2\alpha$ we have

$$\sum_{t=1}^{k-1} (L + t\alpha)^{\frac{\mu}{\alpha}-2} \leq \int_0^{k-1} (L + t\alpha)^{\frac{\mu}{\alpha}-2} dt.$$

Equality (**) holds since

$$\int_0^{k\mu} (L + t)^{-1} \frac{1}{\mu} dt = \frac{1}{\mu} [\log(L + t)]_{t=0}^{k\mu} = \frac{\log(L + k\mu) - \log(L)}{\mu}$$

and

$$\int_0^{k\alpha} (L+t)^{\frac{\mu}{\alpha}-2} \frac{1}{\alpha} dt = \frac{1}{\alpha} \left[\frac{(L+t)^{\frac{\mu}{\alpha}-1}}{\frac{\mu}{\alpha}-1} \right]_{t=0}^{k\alpha} = \frac{(L+k\alpha)^{\frac{\mu}{\alpha}-1} - L^{\frac{\mu}{\alpha}-1}}{\mu - \alpha}$$

for $\alpha < \mu$.

To finish the proof, let us now consider the special case when $\alpha = \frac{\mu}{2}$ (in other words $L_t = L + t\frac{\mu}{2}$). Note that we have

$$\frac{\Gamma_\alpha(L - \mu + \alpha)}{\Gamma_\alpha(L)} = \frac{\Gamma_\alpha(L - \alpha)}{\Gamma_\alpha(L)} = \frac{1}{L - \alpha} = \frac{1}{L - \frac{\mu}{2}}.$$

Thus, according to (48) and (50) we have

$$\begin{aligned} C_k &\stackrel{(48)}{\geq} 1 + \frac{1}{L - \frac{\mu}{2}} \frac{(L - m_\mu + (k-1)\frac{\mu}{2})^2 - (L - m_\mu)^2}{\mu} = 1 + \frac{(L + (k-2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2}{(L - \frac{\mu}{2})\mu} \\ &= \frac{(L + (k-2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2 + (L - \frac{\mu}{2})\mu}{(L - \frac{\mu}{2})\mu} \end{aligned} \quad (51)$$

$$\sum_{t=0}^{k-1} \frac{c_t}{L_t} \stackrel{(50)}{\leq} \frac{1}{L} + \frac{1}{L - \frac{\mu}{2}} \frac{(L + k\frac{\mu}{2})^1 - L^1}{\frac{\mu}{2}} = \frac{1}{L} + \frac{k}{L - \frac{\mu}{2}}. \quad (52)$$

Combining (51), (52) with Theorem 5.4 we obtain

$$\begin{aligned} \sum_{t=1}^k \frac{c_{t-1}}{C_k} \mathbf{E}[f(x_t) - f(x_*)] &\leq \frac{(L - \mu)D_h(x_*, x_0)}{\frac{(L + (k-2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2 + (L - \frac{\mu}{2})\mu}{(L - \frac{\mu}{2})\mu}} + \sigma^2 \frac{\frac{1}{L} + \frac{k}{L - \frac{\mu}{2}}}{\frac{(L + (k-2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2 + (L - \frac{\mu}{2})\mu}{(L - \frac{\mu}{2})\mu}} \\ &= \frac{(L - \mu)(L - \frac{\mu}{2})\mu D_h(x_*, x_0) + \sigma^2 \mu (1 - \frac{\mu}{2L} + k)}{(L + (k-2)\frac{\mu}{2})^2 - (L - \frac{\mu}{2})^2 + (L - \frac{\mu}{2})\mu} \end{aligned}$$

which concludes the proof. \square

C Notation Glossary

Standard		
\mathbb{R}	set of real numbers	
\mathbb{R}_+^n	set of positive vectors in \mathbb{R}^n	
\mathbf{E}	Expectation	
\mathbf{P}	Probability	
\log	natural logarithm	
$\langle \cdot, \cdot \rangle$	Euclidean inner product	
$\ \cdot \ $	standard Euclidean norm	
$D_h(x, y)$	Bregman distance between x, y	(4)
Γ_a	generalization of the Gamma function	(38)
Global		
f	objective to be minimized over set $Q \subseteq \mathbb{R}^n$	(1)
x_*	minimizer of f over Q	(1)
$\nabla f(x)$	gradient of f at x	
h	reference function, f is rel-smooth with respect to h	(5)
L	smoothness parameter (f is L -smooth relative to h)	(5)
μ	strong convexity parameter (f is μ -strongly convex relative to h)	(8)
$x^{(t+1,*)}$	next iterate from Algorithm 1	
$x^{(i)}$	i -th coordinate of $x \in \mathbb{R}^n$	
$\mathbf{1}$	n dimensional vector of ones	
$\mathbf{1}^i$	i -th column of $n \times n$ identity matrix	
relRCD (Section 3)		
τ	minibatch size	
$\alpha(h)$	symmetry measure	(11)
\hat{S}	a random subset of $\{1, 2, \dots, n\}$	
p_0	scalar such that $\mathbf{P}(i \in \hat{S}) = p_0$ for all $i = 1, 2, \dots, n$	
v	parameter vector for ESO	(15)
w	parameter vector for strong convexity	(14)
Δ	$\min_i w^{(i)}/v^{(i)}$	(14)
relSGD (Section 5)		
L_t	stepsize controlling parameter	
c_t	technical tool for analysis	
σ^2	global bound on $L_t \mathbf{E} [\langle \nabla f(x_t) - \tilde{g}_t, x_{t+1} - x_{t+1} \rangle \mid x_t]$	(26)
α	increase rate of L_t in Lemma 5.7	

Table 1: Summary of frequently used notation.