Random extrapolation for primal-dual coordinate descent

Ahmet Alacaoglu [1] **Olivier Fercoq** [2] **Volkan Cevher** [1]

Abstract

We introduce a randomly extrapolated primal-dual coordinate descent method that adapts to sparsity of the data matrix and the favorable structures of the objective function. Our method updates only a subset of primal and dual variables with sparse data, and it uses large step sizes with dense data, retaining the benefits of the specific methods designed for each case. In addition to adapting to sparsity, our method attains fast convergence guarantees in favorable cases *without any modifications*. In particular, we prove linear convergence under metric subregularity, which applies to strongly convex-strongly concave problems and piecewise linear quadratic functions. We show almost sure convergence of the sequence and optimal sublinear convergence rates for the primal-dual gap and objective values, in the general convex-concave case. Numerical evidence demonstrates the state-of-the-art empirical performance of our method in sparse and dense settings, matching and improving the existing methods.

1. Introduction

In this paper, we consider the problem

$$\min_{x\in\mathcal{X}} f(x) + g(x) + h(Ax), \tag{1}$$

where $f, g\colon \mathcal{X} \to \mathbb{R}\cup\{+\infty\}$ and $h\colon \mathcal{Y} \to \mathbb{R}\cup\{+\infty\}$ are proper, lower semicontinuous, convex functions, $A\colon \mathcal{X} \to \mathcal{Y}$ is a linear operator. $\mathcal{X}$ and $\mathcal{Y}$ are Euclidean spaces such that $\mathcal{X} = \prod_{i=1}^n \mathcal{X}_i$, and $\mathcal{Y} = \prod_{j=1}^m \mathcal{Y}_j$. Moreover, f is assumed to have coordinatewise Lipschitz continuous gradients and g, h admit easily computable proximal operators.

Problem (1) is a general template that covers many problems in different fields, such as regularized empirical risk minimization (Shalev-Shwartz & Zhang, 2013; Zhang & Xiao, 2017), optimization with large number of constraints (Patrascu & Necoara, 2017; Fercoq et al., 2019), and total variation (TV) regularized problems (Chambolle et al., 2018; Fercoq & Bianchi, 2019).

The classic choice for solving problem (1) is to use primal-dual methods (Chambolle & Pock, 2011; Vũ, 2013; Condat, 2013). These methods utilize the proximal operators for g, h^* and gradient of the differentiable component f. Randomized versions that we refer to as primal dual coordinate descent (PDCD), are proposed in several works (Zhang & Xiao, 2017; Dang & Lan, 2014; Gao et al., 2019; Fercoq & Bianchi, 2019; Chambolle et al., 2018; Latafat et al., 2019).

First advantage of coordinate-based methods is that they access to blocks of A and update a subset of variables, resulting in cheap per iteration costs. Moreover, they utilize larger step sizes depending on the properties of the problem in selected blocks.

Existing PDCD methods fail to retain both these advantages, as sparsity of A varies. In particular, methods that have cheap per-iteration costs with sparse A (Fercoq & Bianchi, 2019; Latafat et al., 2019), are restricted to use small step sizes with dense A. On the other hand, methods that can use large step sizes with dense A (Chambolle et al., 2018), have high per-iteration costs with sparse A.

Contributions. In this paper, we identify random extrapolation as the key to design a method that combines the benefits of the methods in two camps and propose the primal-dual method with random extrapolation and coordinate descent (PURE-CD). PURE-CD exhibits the advantages of (Fercoq & Bianchi, 2019; Latafat et al., 2019) in the sparse setting and the advantages of (Chambolle et al., 2018) in the dense setting simultaneously, achieving the best of both worlds. As PURE-CD has the favorable properties in both ends of the spectrum, it has the best performance in the regime in between: moderately sparse data. Table 1 compiles a summary for the comparison of PURE-CD and previous methods.

In addition to adapting to the sparsity of A, we prove that PURE-CD also adapts to unknown structures in the problem, and obtains linear rate of convergence, without any modifications in the step sizes. Our linear convergence results apply to strongly convex-strongly concave problems, linear programs, and problems with piecewise linear quadratic

[1] EPFL, Switzerland [2] LTCI, Télécom Paris, Institut Polytechnique de Paris, France. Correspondence to: Ahmet Alacaoglu <ahmet.alacaoglu@epfl.ch>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

	Step sizes with dense data	per iteration cost	block-wise Lipschitz	probability law	Efficient implementation
(Chambolle et al., 2018)	$n\tau_i\sigma\|A_i\|^2 < 1$	m	N/A	arbitrary	direct†
(Fercoq & Bianchi, 2019)	$n^2\tau_i\sigma\|A_i\|^2 < 1$	$\|J(i)\|^*$	Yes	uniform	direct or duplication
(Latafat et al., 2019)	$n^2\tau_i\sigma\|A_i\|^2 < 1$	$\|J(i)\|^*$	No	arbitrary	duplication‡
PURE-CD	$n\tau_i\sigma\|A_i\|^2 < 1$	$\|J(i)\|^*$	Yes	arbitrary	direct

Table 1. Comparison of primal-dual coordinate descent methods. Note that we only compare here the most related methods to ours and include a comprehensive review of other existing methods with comparison to PURE-CD in Section 5. In the last column, we refer to the way one needs to implement the algorithm, for it to be efficient in both sparse and dense settings. $^*J(i)$ is defined in (2). †SPDHG only has implementation for dense setting and not for sparse. ‡The concept of duplication for PDCD is described in (Fercoq & Bianchi, 2019).

functions, involving Lasso, support vector machines and linearly constrained problems with piecewise linear-quadratic objectives. In the general convex case, we prove that the iterates of PURE-CD converges almost surely to a solution of problem (1). Moreover, we show that in this case, the ergodic sequence obtains the optimal $\mathcal{O}(1/k)$ sublinear rate of convergence.

2. Preliminaries

2.1. Notation

For a positive definite matrix V, we denote $\|x\|_V^2 = \langle x, Vx\rangle$. We define the distance of a point x from a set $\mathcal{X}$ as $\operatorname{dist}(x,\mathcal{X}) = \min_{u\in\mathcal{X}} \|u-x\|^2$. Given an index $i \in \{1,\dots,n\}$, the corresponding coordinate of the gradient vector is $\nabla_i f(x)$ and the corresponding coordinate of a vector $x\in\mathcal{X}$ is x^i. Graph of mapping F is denoted by $\operatorname{gra} F$. Recall that $\mathcal{X} = \prod_{i=1}^n \mathcal{X}_i$, and $\mathcal{Y} = \prod_{j=1}^m \mathcal{Y}_j$ and $\mathcal{Z} = \mathcal{X}\times\mathcal{Y}$. For $u \in \mathcal{X}_i$, $U_i(u) \in \mathcal{X}$ is such that each element of $U_i(u)$ is 0, except the block i which contains u. We denote the indicator function of a set $\mathcal{X}$ as $\delta_{\mathcal{X}}$.

Proximal operator with a positive definite V is defined as

$$\operatorname{prox}_{V,g}(x) = \arg\min_u g(u) + \frac{1}{2}\|u-x\|^2_{V^{-1}}.$$

We will need the following notation for the sparse setting,

$$\begin{aligned} J(i) &= \{j \in \{1,\dots,m\} \colon A_{j,i} \neq 0\} \\ I(j) &= \{i \in \{1,\dots,n\} \colon A_{j,i} \neq 0\}. \end{aligned} \tag{2}$$

Given a matrix A and $i\in\{1,\dots,n\}$, $J(i)$ denotes the row indices that correspond to nonzero values in the column indexed by i. Similarly, with $j\in\{1,\dots,m\}$, $I(j)$ gives the column indices corresponding to nonzero values in the row indexed by j.

Moreover, given positive probabilities $(p_i)_{1\le i\le n}$, we define

$$\pi_j = \sum_{i\in I(j)} p_i. \tag{3}$$

In the simple case of $p_i = 1/n$, it is easy to see that $n\pi_j$ corresponds to number of nonzeros in the row indexed by j.

At iteration k, the algorithm randomly picks an index $i_{k+1} \in \{1,\dots,n\}$. To govern the selection rule, we define the probability matrix $P = \operatorname{diag}(p_1,\dots,p_n)$, where $p_i = \mathbb{P}(i_{k+1} = i)$, and $\underline{p} = \min_i p_i$. We define as $\mathcal{F}_k$ the filtration generated by the random indices $\{i_1,\dots,i_k\}$.

Denoting $z = (x,y)$, we define the functions

$$\begin{aligned} D_p(x_{k+1}; z) &= f(x_{k+1}) + g(x_{k+1}) - f(x) - g(x) \\ &\quad + \langle A^\top y, x_{k+1} - x\rangle, \\ D_d(\bar{y}_{k+1}; z) &= h^*(\bar{y}_{k+1}) - h^*(y) - \langle Ax, \bar{y}_{k+1} - y\rangle. \end{aligned}$$

2.2. Optimality

Problem (1) has the following saddle point formulation

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x) + g(x) + \langle Ax, y\rangle - h^*(y).$$

Karush-Kuhn-Tucker (KKT) conditions state that the vector $z_\star = (x_\star, y_\star)$ is a primal-dual solution of the problem when

$$0 \in \begin{bmatrix} \nabla f(x_\star) + \partial g(x_\star) + A^\top y_\star \\ Ax_\star - \partial h^*(y_\star) \end{bmatrix} =: F(z_\star). \tag{4}$$

We call $\mathcal{Z}_\star$ the set of such solutions.

2.3. Metric subregularity

We utilize the metric subregularity assumption for proving linear convergence. This assumption has been used in primal-dual optimization literature for both deterministic (Liang et al., 2016) and randomized algorithms (Latafat et al., 2019; Alacaoglu et al., 2019).

Definition 1. *A set valued mapping $F\colon \mathcal{X} \rightrightarrows \mathcal{Y}$ is metrically subregular at $\bar{x}$ for $\bar{y}$, with $(\bar{x},\bar{y}) \in \operatorname{gra} F$, if there exists $\eta > 0$ with a neighborhood of regularity $\mathcal{N}(\bar{x})$ such that*

$$\operatorname{dist}(x, F^{-1}\bar{y}) \le \eta \operatorname{dist}(\bar{y}, Fx), \quad \forall x \in \mathcal{N}(\bar{x}).$$

We will be interested in the metric subregularity of KKT operator F (see (4)) for 0. Intuitively speaking, as $0 \in F(z_\star), \forall z^\star \in \mathcal{Z}_\star$, metric subregularity of F for 0 essentially gives us a way to characterize the behavior of the iterates around the solution set.

Even though Definition 1 looks daunting, fortunately, one does not need to check it for a given problem. Metric subregularity is well-studied in the literature and it is known to be satisfied in the following cases:

Example 1.
▷ *If $f+g$ and h^* are strongly convex, Definition 1 holds with $\mathcal{N}(\bar{x}) = \mathbb{R}^d$ (Latafat et al., 2019, Lemma IV.2).*
▷ *If f, g, h are piecewise linear quadratic (PLQ) functions, Definition 1 holds with any bounded neighborhood $\mathcal{N}(\bar{x})$ (Latafat et al., 2019, Lemma IV.4).*

PLQ functions include ℓ_1 norm, hinge loss, indicator of polyhedral sets. Thus, second bullet point apply to Lasso, support vector machines, elastic net, and linearly constrained problems with PLQ loss functions (Latafat et al., 2019).

We now state our main assumptions which are standard in the literature (Fercoq & Bianchi, 2019; Chambolle et al., 2018; Latafat et al., 2019; Bauschke & Combettes, 2011):

Assumption 1.
▷ f, g and h are proper, lower semicontinuous, convex.
▷ g is separable, i.e., $g(x) = \sum_{i=1}^n g_i(x^i)$, and f has coordinatewise Lipschitz gradients such that $\forall x \in \mathcal{X}, \forall u \in \mathcal{X}_i$,

$$f(x + U_i(u)) \leq f(x) + \langle \nabla_i f(x), u \rangle + \frac{\beta_i}{2}\|u\|^2. \qquad (5)$$

▷ Set of solutions to problem (1), defined in (4) is nonempty.
▷ Slater's condition holds, which states that $0 \in \operatorname{ri}(\operatorname{dom} h - A \operatorname{dom} g)$ where ri denotes the relative interior.

3. Algorithm

In this section, we will sketch the main ideas behind our algorithm. Primal-dual method[1], due to (Chambolle & Pock, 2011; Condat, 2013; Vũ, 2013) reads as

$$\begin{aligned} \bar{x}_{k+1} &= \operatorname{prox}_{\tau,g}\left(\bar{x}_k - \tau\left(\nabla f(\bar{x}_k) + A^\top \bar{y}_k\right)\right) \\ \bar{y}_{k+1} &= \operatorname{prox}_{\sigma,h^*}\left(\bar{y}_k + \sigma A(2\bar{x}_{k+1} - \bar{x}_k)\right). \end{aligned} \qquad (6)$$

The main intuition behind PDCD methods proposed by (Zhang & Xiao, 2017; Fercoq & Bianchi, 2019; Chambolle et al., 2018) is to incorporate coordinate based updates. Among these methods, (Zhang & Xiao, 2017) specializes in strongly convex-strongly concave problems, whereas the other other ones focus on more general classes of problems.

A closely related approach concentrated on the following interpretation of primal-dual method (6) which is named as TriPD in (Latafat et al., 2019, Algorithm 1)

$$\begin{aligned} \bar{y}_{k+1} &= \operatorname{prox}_{\sigma,h^*}\left(\hat{y}_k + \sigma A\bar{x}_k\right) \\ \bar{x}_{k+1} &= \operatorname{prox}_{\tau,g}\left(\bar{x}_k - \tau\left(\nabla f(\bar{x}_k) + A^\top \bar{y}_{k+1}\right)\right) \\ \hat{y}_{k+1} &= \bar{y}_{k+1} + \sigma A(\bar{x}_{k+1} - \bar{x}_k). \end{aligned} \qquad (7)$$

We notice that by moving the $\bar{y}_{k+1}$ update in TriPD to take place after $\hat{y}_{k+1}$ update, one obtains (6).

As observed in (Latafat et al., 2019), this particular interpretation of primal-dual method is useful for randomization. TriPD-BC as proposed in (Latafat et al., 2019) iterates as

$$\begin{aligned} &\bar{y}_{k+1} = \operatorname{prox}_{\sigma,h^*}\left(y_k + \sigma A x_k\right) \\ &\bar{x}_{k+1} = \operatorname{prox}_{\tau,g}\left(x_k - \tau\left(\nabla f(x_k) + A^\top \bar{y}_{k+1}\right)\right) \\ &\hat{y}_{k+1} = \bar{y}_{k+1} + \sigma A(\bar{x}_{k+1} - x_k) \\ &\text{Draw an index } i_{k+1} \in \{1, \dots, n\} \text{ randomly.} \\ &x_{k+1}^{i_{k+1}} = \bar{x}_{k+1}^{i_{k+1}}, \quad x_{k+1}^j = x_k^j, \forall j \neq i_{k+1} \\ &y_{k+1}^j = \hat{y}_{k+1}^j, \forall j \in J(i_{k+1}), \quad y_{k+1}^j = y_k^j, \forall j \notin J(i_{k+1}). \end{aligned}$$

One immediate limitation of TriPD-BC is that to update y_{k+1}, one needs to know $\bar{x}_{k+1}$, whereas only $\bar{x}_{k+1}^{i_{k+1}}$ is needed to update x_{k+1}. As also discussed in (Latafat et al., 2019), this scheme is suitable when A has special structure such as sparsity. When A is dense, one needs to update all elements of y_{k+1} and $\hat{y}_{k+1}$, in which case one needs to compute both $\bar{y}_{k+1}$ and $\bar{x}_{k+1}$ which has the same cost as a deterministic algorithm.

In the dense setting, for an efficient implementation, one can use duplication of dual variables as described in (Fercoq & Bianchi, 2019). However, in this case one is restricted to use small step sizes as discussed in (Fercoq & Bianchi, 2019). Compared to SPDHG in (Chambolle et al., 2018), the step sizes can be n times worse, deteriorating the performance of the method considerably in the dense setting.

On the other hand, the drawback of SPDHG is that it needs to update all dual variables at every iteration, whereas the methods in (Fercoq & Bianchi, 2019; Latafat et al., 2019) update only a subset of dual variables depending on the sparsity of A. When the dual dimension is high, per iteration cost of (Chambolle et al., 2018) becomes prohibitive.

Our idea, inspired by (Chambolle et al., 2018), to make TriPD-BC efficient for dense setting is to use x_{k+1} rather than $\bar{x}_{k+1}$ in the update of $\hat{y}_{k+1}$. Although simple to state, this modification makes $\hat{y}_{k+1}$ random, rendering the analysis of (Latafat et al., 2019) and other analyses based on monotone operator theory not applicable.

This leads to our algorithm, primal-dual method with random extrapolation and coordinate descent (PURE-CD). Our method uses large step sizes as in (Chambolle et al., 2018) in the dense setting, while staying efficient in terms of per

[1] This method is also known as Vũ-Condat algorithm.

iteration costs in the sparse setting as in (Fercoq & Bianchi, 2019; Latafat et al., 2019); leading to the first general PDCD algorithm that obtains favorable properties in both sparse and dense settings.

Algorithm 1 Primal-dual method with random extrapolation and coordinate descent (PURE-CD)

```
Input: Diagonal matrices $\theta, \tau, \sigma > 0$, chosen according to (8), (9).
for $k = 0, 1 \dots$ do
    $\bar{y}_{k+1} = \mathrm{prox}_{\sigma, h^*}\left(y_k + \sigma A x_k\right)$
    $\bar{x}_{k+1} = \mathrm{prox}_{\tau, g}\left(x_k - \tau\left(\nabla f(x_k) + A^\top \bar{y}_{k+1}\right)\right)$
    Draw $i_{k+1} \in \{1, \dots, n\}$ with $\mathbb{P}(i_{k+1} = i) = p_i$
    $x_{k+1}^{i_{k+1}} = \bar{x}_{k+1}^{i_{k+1}}$
    $x_{k+1}^j = x_k^j, \forall j \neq i_{k+1}$
    $y_{k+1}^j = \bar{y}_{k+1}^j + \sigma_j \theta_j (A(x_{k+1} - x_k))_j, \forall j \in J(i_{k+1})$,
    $y_{k+1}^j = y_k^j, \forall j \notin J(i_{k+1})$
end for
```

4. Convergence Analysis

In this section, we analyze the convergence behavior of Algorithm 1 under various assumptions. We first start with a lemma analyzing one iteration of the algorithm.

Lemma 1. *Let Assumption 1 hold. Recall the definitions of D_p and D_d from Section 2.1 and let $\theta = \mathrm{diag}(\theta_1, \dots, \theta_m)$ and $\pi = \mathrm{diag}(\pi_1, \dots, \pi_m)$ be chosen as*

$$\theta_j = \frac{\pi_j}{\underline{p}}, \textit{ where } \pi_j = \sum_{i \in I(j)} p_i, \textit{ and } \underline{p} = \min_i p_i. \tag{8}$$

We define the functions, given z,

$$V(z) = \frac{\underline{p}}{2}\|x\|^2_{\tau^{-1}P^{-1}} + \frac{\underline{p}}{2}\|y\|^2_{\sigma^{-1}\pi^{-1}},$$
$$\tilde{V}(z) = \frac{\underline{p}}{2}\|x\|^2_{C(\tau)} + \frac{\underline{p}}{2}\|y\|^2_{\sigma^{-1}},$$

where $C(\tau)_i = \frac{2p_i}{\underline{p}\tau_i} - \frac{1}{\tau_i} - p_i \sum_{j=1}^m \pi_j^{-1}\sigma_j\theta_j^2 A_{j,i}^2 - \frac{\beta_i p_i}{\underline{p}}$.

Then, for the iterates of Algorithm 1, $\forall z \in \mathcal{Z}$, it holds that:

$$\mathbb{E}_k\left[D_p(x_{k+1}; z)\right] + \underline{p} D_d(\bar{y}_{k+1}; z) + \mathbb{E}_k\left[V(z_{k+1} - z)\right]$$
$$\leq (1 - \underline{p}) D_p(x_k; z) + V(z_k - z) - \tilde{V}(\bar{z}_{k+1} - z_k).$$

The main technical challenge in the proof of the lemma, compared to the corresponding results in (Latafat et al., 2019) and (Chambolle et al., 2018) is handling stochasticity in both variables x_{k+1}, y_{k+1} (and also $\hat{y}_{k+1}$ for (Latafat et al., 2019)). Using coordinatewise Lipschitz constants of f with arbitrary sampling also requires an intricate analysis.

The result of Lemma 1 is promising for deriving convergence results for Algorithm 1. When $z = z_\star$ in Lemma 1, as $D_p(x_{k+1}; z_\star) \geq 0$, $D_d(\bar{y}_{k+1}; z_\star) \geq 0$ and when step sizes are chosen such that $\tilde{V}$ is a squared norm, Lemma 1 describes a stochastic monotonicity property similar to (Fercoq & Bianchi, 2019). In particular, it shows that $D_p(x_{k+1}; z_\star) + V(z_{k+1} - z_\star)$ which measures the distance to solution in a Bregman distance sense, is monotonically nonincreasing in expectation.

4.1. Almost sure convergence

Almost sure convergence is a fundamental property for randomized methods describing the limiting behavior of the iterates in different realization of the algorithm. The following theorem states that the iterates of Algorithm 1 converge almost surely to a point in the solution set.

Theorem 1. *Let Assumption 1 hold and let θ, π be as in Lemma 1. Choose step sizes τ, σ such that*

$$\tau_i < \frac{2p_i - \underline{p}}{\beta_i p_i + \underline{p}^{-1} p_i \sum_{j=1}^m \pi_j \sigma_j A_{j,i}^2}. \tag{9}$$

The iterates z_k are produced by Algorithm 1. Then, almost surely, there exist $z_\star \in \mathcal{Z}_\star$ such that $z_k \to z_\star$.

We analyze the step size rule (9) in Theorem 1 and compare with existing efficient methods in dense and sparse settings.

Remark 1.
▷ Let A be dense, with all its elements being nonzero, $p_i = 1/n$ and $f(\cdot) = 0$, then the step size rule reduces to

$$\tau_i < \frac{1}{n\sigma\|A_i\|^2},$$

which is the step size rule of SPDHG (Chambolle et al., 2018; Alacaoglu et al., 2019), which is shown to be favorable in the dense setting. In contrast, step size rules of (Fercoq & Bianchi, 2019; Latafat et al., 2019) are n times worse due to duplication, in this case.

▷ Let A be diagonal, and we use $p_i = \frac{1}{n}$, which results in $\pi_j = \frac{1}{n}$. Then,

$$\tau_i < \frac{1}{\beta_i + \sum_{j=1}^m \sigma_j A_{j,i}^2},$$

which is the step size rule of Vu-Condat-CD (Fercoq & Bianchi, 2019), upon using the definition of $J(i)$ from (2). Similarly, Algorithm 1 updates 1 dual coordinate and 1 primal coordinate, in this case. In contrast, SPDHG (Chambolle et al., 2018) updates m dual coordinates, resulting in m times higher per iteration cost.

We note that the step sizes of TriPD-BC (Latafat et al., 2019) depend on global Lipschitz constant of f rather than the coordinatewise ones. Using coordinatewise Lipschitz constants in practice is very important for the success of coordinate descent, as they give larger step sizes (Nesterov, 2012; Richtárik & Takáč, 2014; Fercoq & Richtárik, 2015).

The takeaway from Remark 1 is that Algorithm 1 recovers the characteristics of the best performing methods in fully dense and fully sparse settings. Moreover, as it is the only method with the desirable dependencies in both cases, it has the best properties in the moderate sparse cases. We validate this observation with numerical experiments in Section 6.

4.2. Linear convergence

Linear convergence of primal-dual methods in practice is a widely observed phenomenon (Chambolle & Pock, 2011; Liang et al., 2016). We show that Algorithm 1 also shares this property and obtains linear convergence under metric subregularity, without any modification on the algorithm.

We define the Bregman-type projection onto the solution set

$$z_k^\star = \arg\min_{u \in \mathcal{Z}_\star} D_p(x_k; u) + V(z_k - u). \tag{10}$$

We now show that $z_k^\star$ is well-defined under our assumptions. First, the solution set is convex and closed. Second, $D_p(x_k; u) \geq 0$ for all $u \in \mathcal{Z}_\star$ and it is also lower semicontinuous. Third, we remark that $V(z_k - u)$ is a squared norm (see Lemma 1), thus coercive, therefore the sum is coercive and lower semicontinuous over $\mathcal{Z}_\star$. Hence, $z_k^\star$ exists.

The definition of $z_k^\star$ in (10) is more involved compared to the corresponding quantity in (Latafat et al., 2019). This is in fact due to us using coordinatewise Lipschitz constants in our step sizes, rather than the global Lipschitz constant in (Latafat et al., 2019).

Assumption 2.
KKT operator F is metrically subregular at all $z_\star \in \mathcal{Z}_\star$ for 0, and $\bar{z}_k \in \mathcal{N}(z_\star), \forall z_\star, \forall k$.

Theorem 2. *Let Assumptions 1 and 2 hold. Let θ and the step sizes τ, σ be chosen according to (8) and (9), respectively. Moreover, $z_k^\star = (x_k^\star, y_k^\star)$ is as defined in (10). Then, for z_k generated by Algorithm 1, it follows that*

$$\mathbb{E}\left[\frac{\underline{p}}{2}\|x_k - x_k^\star\|^2_{\tau^{-1}P^{-1}} + \frac{\underline{p}}{2}\|y_k - y_k^\star\|^2_{\sigma^{-1}\pi^{-1}}\right] \\ \leq (1-\rho)^k \Delta_0,$$

where $\rho = \min\left(\underline{p}, \frac{C_{2,\tilde{V}}}{C_{V,2}((2+2c)+(1+c)(\eta\|H-M\|+\bar{\beta}))^2}\right)$, $\Delta_0 = D_p(x_0; z_0^\star) + V(z_0 - z_0^\star)$, *$\bar{\beta}$ is the global Lipschitz constant of f,*
$C_{2,\tilde{V}} = \frac{\underline{p}}{2} \min\left\{\min_i C(\tau)_i, \min_j \sigma_j^{-1}\right\}$,
$C_{V,2} = \frac{1}{2}\max\left\{\max_i \frac{1}{\tau_i}, \max_j \frac{1}{\sigma_j}\right\}$, $c = C_{2,V}\sqrt{\|A\|/2}$,
$C_{2,V} = \sqrt{\frac{2}{\underline{p}\min\left\{\min_i \tau_i^{-1}p_i^{-1}, \min_j \sigma_j^{-1}\pi_j^{-1}\right\}}}$, *and*

$$H = \begin{bmatrix} \tau^{-1} & A^\top \\ 0 & \sigma^{-1} \end{bmatrix}, \quad M = \begin{bmatrix} 0 & A^\top \\ -A & 0 \end{bmatrix}.$$

The first remark about Theorem 2 is that since metric subregularity constant η is not required in the algorithm, the step sizes to achieve linear convergence are the same step sizes as (9). Therefore, PURE-CD adapts to structures on the problem, without any need to modify the algorithm, and attains linear rate of convergence. This supports the well-known observation that primal-dual algorithms converge linearly on most problems, with standard step sizes in (9).

In particular, a direct corollary of our theorem is that for problems listed in Example 1, PURE-CD obtains linear rate of convergence. For the first two cases in Example 1, our result applies directly since the neighborhood of subregularity $\mathcal{N}(z_\star)$ is the whole space. For the third case, we have to assume additionally that $\bar{z}_k$ is contained in a compact set, since $\mathcal{N}(z_\star)$ is not the whole space, and is bounded. A sufficient assumption for this is when the domains of g and h^* are compact. We note that compactness is only required for this result in our paper. This is common to other results for PDCD methods with metric subregularity (Latafat et al., 2019; Alacaoglu et al., 2019). The issue, as explained in (Alacaoglu et al., 2019), stems from a fundamental limitation of the existing analyses of PDCD methods.

Many results in the literature for linear convergence only applies to the first case in Example 1, when g, h^* are strongly convex (Zhang & Xiao, 2017; Chambolle et al., 2018). Moreover, these results require setting step sizes depending on strong convexity constants of g, h^*, therefore not applicable when strong convexity is absent. Our result applies to more general problems and it uses step sizes independent of these constants. Our algorithm can be directly applied to any problem satisfying Assumption 1 and fast convergence will occur provably, if the selected problem is in Example 1.

Compared with the linear convergence rate in (Latafat et al., 2019) for TriPD-BC, our result have a similar contraction factor, however, due to larger step sizes (see Remark 1), the rate comes with a better constant.

4.3. Ergodic rates

In this section, we study Algorithm 1 in the general case, under Assumption 1, and show the optimal $\mathcal{O}(1/k)$ convergence rate on the ergodic sequence. The quantity of interest is the primal-dual gap function (Chambolle & Pock, 2011)

$$\begin{aligned} G(\bar{x}, \bar{y}) = \sup_{z \in \mathcal{Z}} & f(\bar{x}) + g(\bar{x}) + \langle A\bar{x}, y\rangle - h^*(y) \\ & - f(x) - g(x) - \langle Ax, \bar{y}\rangle + h^*(\bar{y}). \end{aligned} \tag{11}$$

A related quantity is the restricted gap function (Chambolle & Pock, 2011), which, for any set $\mathcal{C} \subset \mathcal{Z}$ is defined as

$$\begin{aligned} G_{\mathcal{C}}(\bar{x}, \bar{y}) = \sup_{z \in \mathcal{C}} & f(\bar{x}) + g(\bar{x}) + \langle A\bar{x}, y\rangle - h^*(y) \\ & - f(x) - g(x) - \langle Ax, \bar{y}\rangle + h^*(\bar{y}). \end{aligned} \tag{12}$$

Due to randomization in PDCD, we are interested in the expected primal-dual gap, denoted as $\mathbb{E}\left[G_{\mathcal{C}}(\bar{x}, \bar{y})\right]$. As noted by Dang & Lan (2014), it is technically challenging to prove rates for this quantity as it is the expectation of supremum. Recently, (Alacaoglu et al., 2019) used a technique to show convergence of expected primal-dual gap for SPDHG of (Chambolle et al., 2018). This rate is for ergodic sequence averaging x_k and the full dual variable $\bar{y}_k$. We can use this technique for our analysis. However, there remains another technical challenge as full dual variable is not computed in PURE-CD. Thus, averaging $\bar{y}_k$ is not feasible in our case.

In addition to Assumption 1, in this section we will assume separability of h, to be able to do an efficient averaging with the dual iterate.

Due to the asymmetric nature of Algorithm 1, there are fundamental difficulties for proving a rate with averaging y_{k+1}. On this front, we propose a new type of analysis for the dual variable. To start with, we define the following iterate which has the same cost to compute as y_{k+1} each iteration. Let $\breve{y}_1 = y_1 = \bar{y}_1$,

$$\begin{aligned} \breve{y}_{k+1}^j &= \bar{y}_{k+1}^j, \quad \forall j \in J(i_{k+1}), \\ \breve{y}_{k+1}^j &= \breve{y}_k^j, \qquad \forall j \notin J(i_{k+1}). \end{aligned} \tag{13}$$

We note that $\breve{y}_k$ is $\mathcal{F}_k$-measurable and more useful properties of $\breve{y}_k$ for analysis are given in Lemma 5 in Appendix B.4.

Due to the definition of $\breve{y}_k$, it is now feasible to compute and average this iterate. We can show the convergence of expected primal-dual gap by averaging $\breve{y}_k$ and x_k. We remark that we use some coarse inequalities to give simple constants for Theorem 3 and Theorem 4. Therefore, the bounds are not optimized with respect to dimension dependence. In Appendix B, we give these theorems with their original, tighter bounds and we show how we transform the tighter bounds into the constants we give in this section.

Theorem 3. *Let Assumption 1 hold and θ, τ, σ are chosen as in (8), (9). Moreover, let h be separable.*

We define $x_K^{av} = \frac{1}{K}\sum_{k=1}^K x_k$ and $y_K^{av} = \frac{1}{K}\sum_{k=1}^K \breve{y}_k$, where $\breve{y}_k$ is defined in (13), then it holds that for any bounded set $\mathcal{C} = \mathcal{C}_x \times \mathcal{C}_y \subset \mathcal{Z}$

$$\mathbb{E}\left[G_{\mathcal{C}}(x_K^{av}, y_K^{av})\right] \le \frac{C_g}{\underline{p}K},$$

where $C_g = \sum_{i=1}^4 C_{g,i}$, $C_{\tau,\tilde{V}} = \min_i C(\tau)_i \tau_i$,
$C_{g,1} = \sup_{z\in\mathcal{C}} \left\{2\underline{p}\|x_0 - x\|^2_{\tau^{-1}P^{-1}} + 2\underline{p}\|y_0 - y\|^2_{\sigma^{-1}\pi^{-1}}\right\} +$
$4\sqrt{\Delta_0 \underline{p}^{-1}}\|A\| \sup_{y\in\mathcal{C}_y} \|y\|_{\tau P}$
$+ 2\sqrt{\Delta_0(\underline{p}^{-1} + 2\underline{p}^{-3}C_{\tau,\tilde{V}}^{-1})}\|A\| \sup_{x\in\mathcal{C}_x} \|x\|_{\sigma\pi}$,
$\sum_{i=2}^4 C_{g,i} = \Delta_0\left(5 + 9\underline{p}^{-1} + C_{\tau,\tilde{V}}^{-1}\left(1 + 10\underline{p}^{-1} + 14\underline{p}^{-2}\right)\right)$
$+ (1-\underline{p})(f(x_0) + g(x_0) - f(x_\star) - g(x_\star)) + h^*(y_0) - h^*(y_\star)$
$+ \underline{p}\|Ax_\star\|^2_{\sigma\pi^{-1}} + \|A^\top y_\star\|^2_{\tau P}$.

Remark 2. When implementing averaging of x_k, and $\breve{y}_k$, one should use a technique similar to (Dang & Lan, 2015). The main idea is to only update the averaged vector at the coordinates where an update occurred. For this, one needs to remember for each coordinate, the last time it is updated, wait until a coordinate is selected again and update the averaged vector using this information.

The result in Theorem 3 would give a rate for primal-dual gap when $\mathcal{C} = \mathcal{Z}$. However, in general such a rate is not desirable as taking a supremum over $\mathcal{Z}$ might result in an infinite bound. This rate would be meaningful when both primal and dual domains are bounded in which case one would take the supremum in $C_{g,1}$ over the bounded domains.

Alternatively, in the following theorem, we show that for two important special cases, we can extend this result to show guarantees without bounded domains. Namely, we show the same rate for the case when $h(\cdot) = \delta_{\{b\}}(\cdot)$, $b \in \mathbb{R}^m$ to cover linearly constrained problems. Moreover, we show the result for the case when h is Lipschitz continuous.

Theorem 4. *Let Assumption 1 hold. We use the same parameters θ, τ, σ and the definitions for x_K^{av} and y_K^{av} as Theorem 3. We consider two cases separately:*
▷ *If $h(\cdot) = \delta_{\{b\}}(\cdot)$, we obtain*

$$\mathbb{E}[f(x_K^{av}) + g(x_K^{av}) - f(x_\star) - g(x_\star)] \le \frac{C_o}{\underline{p}K}.$$

$$\mathbb{E}[\|Ax_K^{av} - b\|] \le \frac{C_f}{\underline{p}K}.$$

▷ *If h is L_h-Lipschitz continuous, we obtain*

$$\begin{aligned} &\mathbb{E}[f(x_K^{av}) + g(x_K^{av}) + h(Ax_K^{av}) \\ &\qquad - f(x_\star) - g(x_\star) - h(Ax_\star)] \le \frac{C_l}{\underline{p}K}, \end{aligned}$$

where $C_f = 3c_1\|x_\star - x_0\|_{\tau^{-1}P^{-1}} + 2\sqrt{c_1 C_s} + 4c_1\|y_\star - y_0\|_{\sigma^{-1}\pi^{-1}}$,
$C_o = C_s + \|y_\star\|_{\sigma^{-1}\pi^{-1}} C_f + 2c_1\underline{p}^{-1}V(z_0 - z_\star)$,
$C_l = C_s + c_1\|x_\star - x_0\|^2_{\tau^{-1}P^{-1}} + 4c_1 L_h^2$,
$C_s = C_{g,2} + C_{g,5} + C_{g,6}$, *with* $c_1 = 2\underline{p} + 2$, $C_{g,2}$ *as defined in the statement of Theorem 3 in Appendix B.4 and* $C_{g,5}, C_{g,6}$ *are defined in the proof in* (79), (80).

5. Related works

In the deterministic setting, many primal-dual methods are proposed (Chambolle & Pock, 2011; Vũ, 2013; Condat, 2013; Tan et al., 2020; Latafat et al., 2019). The standard results in these papers include linear convergence when g, h^* are strongly convex, with step sizes selected by using strong convexity constants. In addition, these papers also show sublinear $\mathcal{O}(1/k)$ rate with general convexity, which is known to be optimal (Nesterov, 2005). Moreover, linear rates under metric subregularity is shown for deterministic methods in (Liang et al., 2016; Latafat et al., 2019).

Randomized coordinate descent is proposed in (Nesterov, 2012) and improved by a large body of subsequent papers (Richtárik & Takáč, 2014; Fercoq & Richtárik, 2015). Primal randomized coordinate descent requires full separability on the nonsmooth parts of the objective function. Nonsmooth and nonseparable functions are handled by primal-dual coordinate descent methods (Fercoq & Bianchi, 2019).

One of the first primal-dual coordinate descent (PDCD) methods is SPDC, which is proposed in (Zhang & Xiao, 2017), that solves a special case of problem (1) with $f = 0$. SPDC has linear convergence when g, h^* are strongly convex and the step sizes are selected according to strong convexity constants. In the general convex case, SPDC has perturbation-based analysis, which needs to set an ϵ, requires knowing $\|x_\star\|^2$, and shows ϵ-based iteration complexity results, and not anytime convergence rates. Almost sure convergence of the iterates of SPDC is not proven in the general convex case. Moreover, the step sizes of SPDC are scalar and they depend on the maximum block norm of A. It is shown in (Zhang & Xiao, 2017) that in the specific cases when $g(x) = \|x\|^2$ or $g(x) = \|x\|_1 + \|x\|^2$, one can use a special implementation for efficiency with sparse data.

Tan et al. (2020) proposed a new method similar to SPDC with the same type of guarantees as (Zhang & Xiao, 2017). Due to similar analysis techniques, this method inherits the abovementioned drawbacks of SPDC. For this method, Tan et al. (2020) showed a new implementation technique for sparse data, that can be used with any separable $g(x)$.

For solving the specific case of empirical risk minimization problems, stochastic dual coordinate ascent (SDCA) is proposed in (Shalev-Shwartz & Zhang, 2013; 2014). SDCA uses strong convexity constant to set step sizes and attain linear convergence. A limitation of SDCA is to require strong convexity in the primal, to ensure smoothness of the dual objective, which is essential in the design of the method.

Another early PDCD method is by (Dang & Lan, 2014) where the authors focused on showing sublinear convergence rates. The authors showed guarantees for a relaxed version of expected primal-dual gap function in (11).

Building on (Dang & Lan, 2014), block-coordinate variants of alternating direction method of multipliers (ADMM) are proposed in (Gao et al., 2019; Xu & Zhang, 2018). These papers focus on linearly constrained problems and show ergodic sublinear convergence rates. Moreover, (Xu & Zhang, 2018) showed that under strong convexity assumption and special decomposition of the blocks, the method achieves linear convergence. This linear convergence result, similar to (Zhang & Xiao, 2017) requires knowing the strong convexity constants to set the algorithmic parameters. Moreover, these results generally set step sizes depending on global Lipschitz constants and norm of whole matrix A.

PDCD variants are also proposed in (Combettes & Pesquet, 2015; 2019; Pesquet & Repetti, 2015) and analyzed under the general setting of monotone operators. These methods use global constants of the problem such as global Lipschitz constant of smooth part and $\|A\|$, rather than blockwise constants, resulting in worse practical performance, as illustrated in the experiments of (Chambolle et al., 2018).

Another early PDCD variant to solve problem (1) in its full generality, where f, g, h are all nonseparable, is by Fercoq & Bianchi (2019). This method uses coordinatewise Lipschitz constants of the smooth part and it is designed to exploit sparsity of A. This method has almost sure convergence guarantees as well as linear convergence when g, h^* are strongly convex. As opposed to most results in this nature, it is not required to know strong convexity constants to set the step sizes. In the general convex case, the method has $\mathcal{O}(1/\sqrt{k})$ rate for a randomly selected iterate. As argued in Section 4.1, main limitation of (Fercoq & Bianchi, 2019) is that small step sizes are required when matrix A is dense. Moreover, the results in this paper are restricted to uniform probability law for selecting coordinates.

One of the most related works to ours, and a building block of PURE-CD is TriPD-BC from (Latafat et al., 2019). The authors showed almost sure convergence of the iterates and linear convergence under metric subregularity, by using global Lipschitz constants of f for the step sizes. This paper did not have any sublinear convergence rates in the general convex case. Similar to (Fercoq & Bianchi, 2019), TriPD-BC is designed for sparse setting and a naive implementation in the dense setting requires the same per iteration cost as the deterministic algorithm. An efficient implementation is by duplication of dual variables, which as explained in (Fercoq & Bianchi, 2019) results in small step sizes (see Section 4.1).

Another building block of PURE-CD is SPDHG by (Chambolle et al., 2018), to solve (1) when $f = 0$. Linear convergence result of SPDHG by (Chambolle et al., 2018) is similar to (Zhang & Xiao, 2017) and requires setting step sizes with strong convexity constants. In the general convex case and partially strongly convex case, (Chambolle et al., 2018) proved optimal sublinear rates. Recently, (Alacaoglu et al., 2019) analyzed SPDHG and proved additional theoretical results. In particular, this work showed almost sure convergence of the iterates of SPDHG and linear convergence under metric subregularity. Even though it is fast in the dense setting, the main limitation of SPDHG, as discussed in Section 4.1 is that it needs to update all the dual coordinates, resulting in high per iterations costs in the sparse setting.

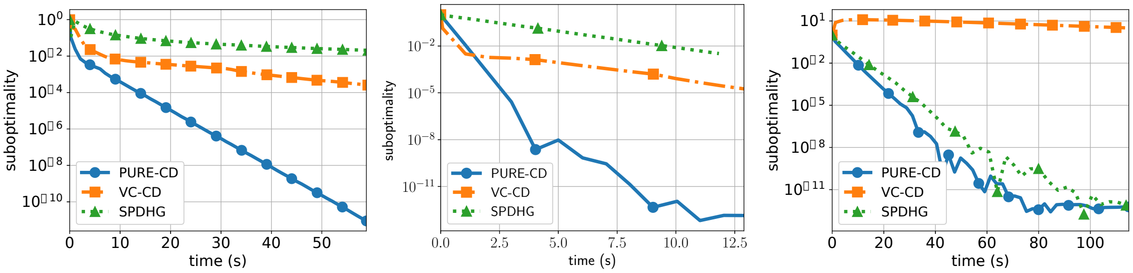

Figure 1. Lasso: Left: rcv1, $n = 20,242, m = 47,236$, density $= 0.16\%$, $\lambda = 10$; Middle: w8a, $n = 49,749, m = 300$, density $= 3.9\%$, $\lambda = 10^{-1}$; Right: covtype, $n = 581,012$, $m = 54$, density $= 22.1\%$, $\lambda = 10$.

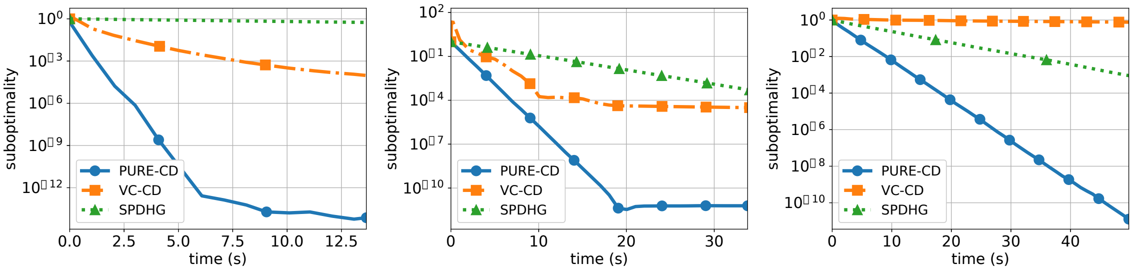

Figure 2. Ridge regression: Left: sector, $n = 6,412, m = 55,197$, density $= 0.3\%$, $\lambda = 0.1$; Middle: a9a, $n = 32,561, m = 123$, density $= 11.3\%$, $\lambda = 0.1$; Right: mnist, $n = 60,000$, $m = 780$, density $= 19.2\%$, $\lambda = 1$.

Other PDCD methods are proposed in (Luke & Malitsky, 2018; Alacaoglu et al., 2017) where the authors focused on linearly constrained problems and proved sublinear rates.

6. Numerical experiments

6.1. Effect of sparsity

As explained in Section 4.1, and Remark 1, PURE-CD brings together the benefits of different methods that are designed for dense and sparse cases. We will now compare the empirical performance of PURE-CD with Vu-Condat-CD from (Fercoq & Bianchi, 2019) which has desirable properties with sparse data and SPDHG from (Chambolle et al., 2018) which has desirable properties with dense data.

We select uniform sampling, $p_i = 1/n$, so (9) simplifies to

$$\tau_i < \frac{1}{\sum_{j=1}^m \theta_j \sigma_j A_{j,i}^2}. \tag{14}$$

We provide a step size policy inspired by the step size rules chosen in (Chambolle et al., 2018) and (Fercoq & Bianchi, 2019). We use the following step sizes, for $\gamma < 1$

$$\sigma_j = \frac{1}{\theta_j \max_{i'} \|A_{i'}\|}, \quad \tau_i = \frac{\gamma \max_{i'} \|A_{i'}\|}{\|A_i\|^2}.$$

We note that in contrast to (Chambolle et al., 2018), step sizes are both diagonal. In our case, it is important to utilize diagonal step sizes for both primal and dual variables since we perform coordinate-wise updates for both primal and dual variables and the step sizes need to be set appropriately to obtain good practical performance. For SPDHG and Vu-Condat-CD, we use step sizes suggested in the papers.

In the edge cases (one nonzero element per row or fully dense), it is easy to see that our step size policy reduces to the suggested step sizes of (Chambolle et al., 2018) and (Fercoq & Bianchi, 2019).

For experiments, we used the generic coordinate descent solver, implemented in Cython, by Fercoq (2019), which includes an implementation of Vu-Condat-CD with duplication and we implemented SPDHG and PURE-CD. We solve Lasso and ridge regression, where we let $g(x) = \lambda\|x\|_1$, $h(Ax) = \frac{1}{2}\|Ax - b\|^2$, $f = 0$, and $g(x) = \frac{\lambda}{2}\|x\|^2$, $h(Ax) = \frac{1}{2}\|Ax - b\|^2$, $f = 0$, respectively, in our template (1). Then, we apply all the methods to the dual problems of these, to access data by rows.

We use datasets from LIBSVM with different sparsity levels (Chang & Lin, 2011). The properties of each data matrix are given in the caption of the corresponding figures. For preprocessing, we removed all-zero rows and all-zero columns of A and we performed row normalization. The results are compiled in Figures 1 and 2.

We observe the behavior predicted by theory. With sparse data such as rcv1, where density level is 0.16%, SPDHG makes very little progress in the given time window. The reason is that the per iteration cost of SPDHG in this case is updating $47,236$ dual variables, whereas for PURE-CD and Vu-Condat-CD, the cost is updating 75 dual variables.

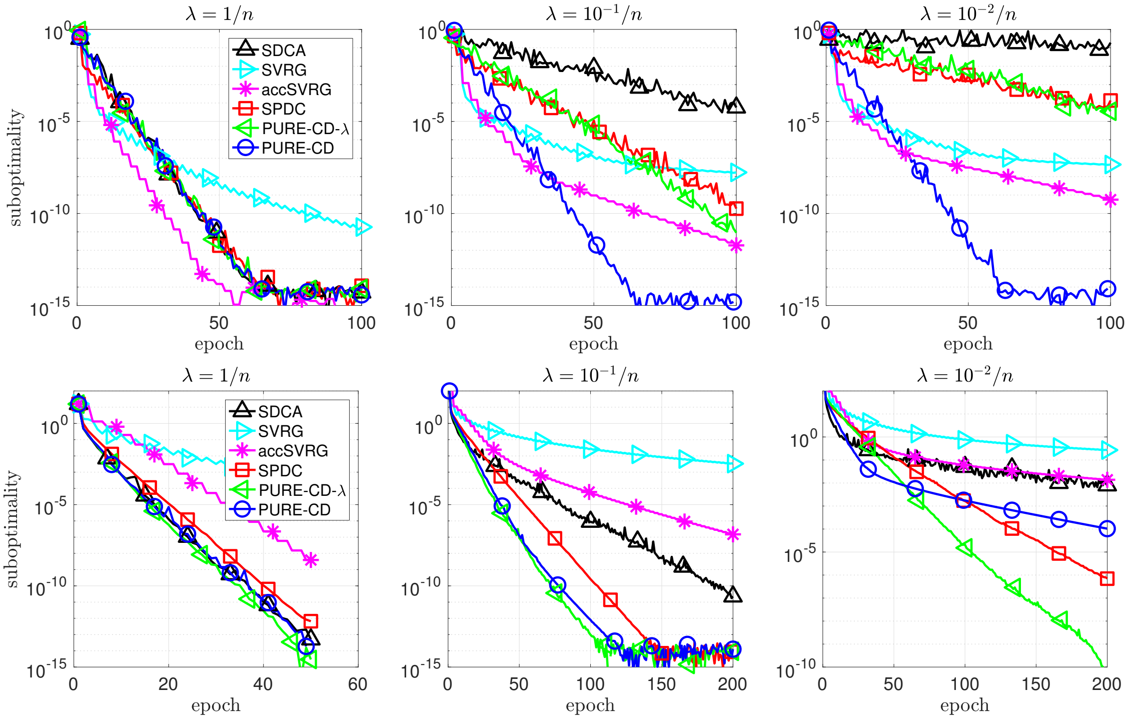

Figure 3. top: a9a, $n = 32,561$, $m = 123$, bottom: sector, $n = 6,412$, $m = 55,197$.

We note that PURE-CD is faster than Vu-Condat-CD due to better step sizes. On the other hand, with moderate sparsity, SPDHG and Vu-Condat-CD is comparable, whereas PURE-CD exhibits the best performance. For denser data, SPDHG and PURE-CD exhibit similar behavior where Vu-Condat-CD is slower than both due to smaller step sizes.

6.2. Comparison with specialized methods

In this section, we compare the practical performance of PURE-CD with state-of-the-art algorithms that are designed for strongly convex-strongly concave problems. Due to space constraints, we defer some of the plots and more details about experiments to the appendix. We focus on the problem $\min_x \frac{1}{n}\sum_{i=1}^n h_i(A_i x) + \frac{\lambda}{2}\|x\|^2$, where $h_i(x) = (x - b_i)^2$. Each h_i is smooth with Lipschitz constants $L_i = 2$ and the second component is strongly convex, which results in strong convexity in both primal and dual problems.

In this case, the algorithms SDCA (Shalev-Shwartz & Zhang, 2013), ProxSVRG (Xiao & Zhang, 2014), Accelerated SVRG (Zhou et al., 2018), SPDC (Zhang & Xiao, 2017) are all designed to use the strong convexity to obtain linear convergence. These algorithms use the strong convexity constant λ for setting the algorithmic parameters. Moreover, as all the abovementioned algorithms have special implementations to exploit sparsity, we make the comparison with respect to number of passes of the data, rather than time. The results are compiled for two datasets in Figure 3 and more datasets are included in Appendix A. We use theoretical step sizes for all the algorithms, given in the respective papers.

- PURE-CD-λ: This variant uses the non-agnostic step sizes, using λ, which still satisfy the theoretical requirement (14).

$$\sigma_j = \frac{n}{\theta_j \sqrt{n\lambda} \max_{i'} \|A_{i'}\|}, \quad \tau_i = \frac{\gamma \sqrt{n\lambda} \max_{i'} \|A_{i'}\|}{n\|A_i\|^2}.$$

- PURE-CD: This variant is with the standard agnostic step sizes.

$$\sigma_j = \frac{n}{\theta_j \max_{i'} \|A_{i'}\|}, \quad \tau_i = \frac{\gamma \max_{i'} \|A_{i'}\|}{n\|A_i\|^2}.$$

We observe that PURE-CD has a consistent linear convergence behavior as predicted by theory. In most of the datasets (see Appendix A), it has the fastest convergence behavior. However, in some datasets, as λ gets smaller, we observed that the linear rate of PURE-CD slowed down, which motivated us to try PURE-CD-λ, which incorporates the knowledge of λ as the other methods. It seems to show favorable behavior when PURE-CD slows down.

The takeaway message is that PURE-CD, which is designed for a general problem, adapts to strong convexity well with agnostic step sizes in most cases. However, in some cases, it does not perform as good as the algorithms which are designed to exploit strong convexity. In those cases however, one can choose separating step sizes of PURE-CD accordingly, and use PURE-CD-λ to get better performance.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no 725594 - time-data) and the Swiss National Science Foundation (SNSF) under grant number 200021_178865/1.

References

Alacaoglu, A., Dinh, Q. T., Fercoq, O., and Cevher, V. Smooth primal-dual coordinate descent algorithms for nonsmooth convex optimization. In *Advances in Neural Information Processing Systems*, pp. 5852–5861, 2017.

Alacaoglu, A., Fercoq, O., and Cevher, V. On the convergence of stochastic primal-dual hybrid gradient. *arXiv preprint arXiv:1911.00799*, 2019.

Arrow, K. J., Azawa, H., Hurwicz, L., and Uzawa, H. *Studies in linear and non-linear programming*, volume 2. Stanford University Press, 1958.

Bauschke, H. H. and Combettes, P. L. *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, 2011.

Chambolle, A. and Pock, T. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40(1):120–145, 2011.

Chambolle, A., Ehrhardt, M. J., Richtárik, P., and Schonlieb, C.-B. Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications. *SIAM Journal on Optimization*, 28(4):2783–2808, 2018.

Chang, C.-C. and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Combettes, P. L. and Pesquet, J.-C. Stochastic quasi-fejér block-coordinate fixed point iterations with random sweeping. *SIAM Journal on Optimization*, 25(2):1221–1248, 2015.

Combettes, P. L. and Pesquet, J.-C. Stochastic quasi-fejér block-coordinate fixed point iterations with random sweeping ii: mean-square and linear convergence. *Mathematical Programming*, 174(1-2):433–451, 2019.

Condat, L. A primal–dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.

Dang, C. and Lan, G. Randomized methods for saddle point computation. *arXiv preprint arXiv:1409.8625*, 3(4), 2014.

Dang, C. D. and Lan, G. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881, 2015.

Fercoq, O. A generic coordinate descent solver for nonsmooth convex optimisation. *Optimization Methods and Software*, pp. 1–21, 2019.

Fercoq, O. and Bianchi, P. A coordinate-descent primal-dual algorithm with large step size and possibly nonseparable functions. *SIAM Journal on Optimization*, 29(1):100–134, 2019.

Fercoq, O. and Richtárik, P. Accelerated, parallel, and proximal coordinate descent. *SIAM Journal on Optimization*, 25(4):1997–2023, 2015.

Fercoq, O., Alacaoglu, A., Necoara, I., and Cevher, V. Almost surely constrained convex optimization. In *International Conference on Machine Learning*, pp. 1910–1919, 2019.

Gao, X., Xu, Y.-Y., and Zhang, S.-Z. Randomized primal–dual proximal block coordinate updates. *Journal of the Operations Research Society of China*, 7(2):205–250, 2019.

Iutzeler, F., Bianchi, P., Ciblat, P., and Hachem, W. Asynchronous distributed optimization using a randomized alternating direction method of multipliers. In *52nd IEEE conference on decision and control*, pp. 3671–3676. IEEE, 2013.

Latafat, P., Freris, N. M., and Patrinos, P. A new randomized block-coordinate primal-dual proximal algorithm for distributed optimization. *arXiv preprint arXiv:1706.02882v4*, 2019.

Liang, J., Fadili, J., and Peyré, G. Convergence rates with inexact non-expansive operators. *Mathematical Programming*, 159(1-2):403–434, 2016.

Luke, D. R. and Malitsky, Y. Block-coordinate primal-dual method for nonsmooth minimization over linear constraints. In *Large-Scale and Distributed Optimization*, pp. 121–147. Springer, 2018.

Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

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

Patrascu, A. and Necoara, I. Nonasymptotic convergence of stochastic proximal point methods for constrained convex optimization. *Journal of Machine Learning Research*, 18: 198–1, 2017.

Pesquet, J.-C. and Repetti, A. A class of randomized primal-dual algorithms for distributed optimization. *Journal of Nonlinear and Convex Analysis*, 16(12):2453–2490, 2015.

Richtárik, P. and Takáč, M. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144 (1-2):1–38, 2014.

Shalev-Shwartz, S. and Zhang, T. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(Feb):567–599, 2013.

Shalev-Shwartz, S. and Zhang, T. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *International Conference on Machine Learning*, pp. 64–72, 2014.

Tan, C., Qian, Y., Ma, S., and Zhang, T. Accelerated dual-averaging primal-dual method for composite convex minimization. *Optimization Methods and Software*, 0(0):1–26, 2020. doi: 10.1080/10556788.2020.1713779.

Tran-Dinh, Q., Fercoq, O., and Cevher, V. A smooth primal-dual optimization framework for nonsmooth composite convex minimization. *SIAM Journal on Optimization*, 28 (1):96–134, 2018.

Vũ, B. C. A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681, 2013.

Xiao, L. and Zhang, T. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

Xu, Y. and Zhang, S. Accelerated primal–dual proximal block coordinate updating methods for constrained convex optimization. *Computational Optimization and Applications*, 70(1):91–128, 2018.

Zhang, Y. and Xiao, L. Stochastic primal-dual coordinate method for regularized empirical risk minimization. *The Journal of Machine Learning Research*, 18(1):2939–2980, 2017.

Zhou, K., Shang, F., and Cheng, J. A simple stochastic variance reduced algorithm with fast convergence rates. In *International Conference on Machine Learning*, pp. 5975–5984, 2018.