Intl. Trans. in Op. Res. xx (2021) 1–24

Adaptive Coordinate Sampling for Stochastic Primal-Dual Optimization

Huikang Liu[†], Xiaolu Wang[*], and Anthony Man-Cho So[*]

[†]*Imperial College Business School, Imperial College London, London, UK*
[*]*Department of Sys. Eng. and Eng. Mgmt, The Chinese University of Hong Kong, Hong Kong, China*
E-mail: huikang.liu@imperial.ac.uk [Liu]; xlwang@se.cuhk.edu.hk [Wang]; manchoso@se.cuhk.edu.hk [So]

Received DD MMMM YYYY; received in revised form DD MMMM YYYY; accepted DD MMMM YYYY

Abstract

We consider the regularized empirical risk minimization (ERM) of linear predictors, which arises in a variety of problems in machine learning and statistics. After reformulating the original ERM as a bilinear saddle-point problem, we can apply stochastic primal-dual methods to solve it. Sampling the primal or dual coordinates with a fixed non-uniform distribution is usually employed to accelerate the convergence of the algorithm, but such a strategy only exploits the global information of the objective function. To capture its local structures, we propose an adaptive importance sampling strategy that chooses the coordinates based on delicately-designed non-uniform and non-stationary distributions. When our adaptive coordinate sampling strategy is applied to the SPDC (Zhang and Xiao, 2017), we prove that the resulting algorithm enjoys linear convergence. Moreover, we show that the ideal form of our adaptive sampling exhibits strictly sharper convergence rate under certain conditions compared with the vanilla SPDC. We also extend our sampling strategy to other algorithms including DSPDC (Yu et al., 2015) and SPD1-VR (Tan et al., 2018), where both the primal and dual coordinates are randomly sampled. Our experiment results show that the proposed strategy significantly improves the convergence performance of the methods when compared with existing sampling strategies.

Keywords: primal-dual methods; stochastic optimization algorithms; adaptive importance sampling; machine learning

1. Introduction

A wide range of problems in machine learning and statistics boil down to the following empirical risk minimization with linear predictor:

$$\min_{x\in\mathbb{R}^d}\left\{P(x)=\frac{1}{n}\sum_{i=1}^{n}\phi_i(a_i^\top x)+g(x)\right\}, \tag{1}$$

where $a_i \in \mathbb{R}^d$ is the i-th feature vector, $\phi_i : \mathbb{R} \to \mathbb{R}$ is the i-th convex closed loss function associated with the linear prediction $a_i^\top x$, and $g : \mathbb{R}^d \to \mathbb{R}$ is a convex regularization function for the predictor $x \in \mathbb{R}^d$. Problem (1) arises in many classification and regression tasks, where ϕ_i and g take different forms, such as

- logistic regression with sigmoid loss:

$$\phi_i(z) = \log(1 + \exp(-b_i z)),\ b_i \in \{\pm 1\}, \tag{2}$$

- support vector machine with smoothed hinge loss:

$$\phi_i(z) = \begin{cases} 0 & \text{if } b_i z \geq 1 \\ 1/2 - b_i z & \text{if } b_i z \leq 0 \\ (1/2)(1 - b_i z)^2 & \text{otherwise} \end{cases},\ b_i \in \{\pm 1\}, \tag{3}$$

- linear/ridge regression with squared loss:

$$\phi_i(z) = \frac{1}{2}(z - b_i)^2,\ b_i \in \mathbb{R}, \tag{4}$$

Two commonly used regularizers include the ℓ_2-norm $g(x) = (\lambda/2)\|x\|_2^2$ where $\lambda > 0$, and the $(\ell_1 + \ell_2)$-norm $g(x) = \lambda_1 \|x\|_1 + (\lambda_2/2)\|x\|_2^2$ where $\lambda_1, \lambda_2 > 0$.

Instead of directly solving the primal problem (1), it is often advantageous to tackle its equivalent primal-dual reformulation (Esser et al., 2010)

$$\min_{x \in \mathbb{R}^d} \max_{y \in \mathbb{R}^n} \left\{ F(x, y) = \frac{1}{n} y^\top A x - \frac{1}{n} \sum_{i=1}^{n} \phi_i^*(y_i) + g(x) \right\}, \tag{5}$$

where $A = [a_1^\top, \ldots, a_n^\top]^\top \in \mathbb{R}^{n \times d}$, $\phi_i^*(y_i) = \sup_{\nu \in \mathbb{R}} \{\nu y_i - \phi_i(\nu)\}$ is the convex conjugate function of ϕ_i. In this paper, we focus on the convex-concave saddle point problem (5). If ϕ_i is $(1/\gamma)$-smooth (i.e., ϕ_i has Lipschitz continuous gradient with constant $1/\gamma$), then its conjugate ϕ_i^* is γ-strongly convex (see Chapter E, Theorem 4.2.2 in Hiriart-Urruty and Lemaréchal (2012)). We consider a decomposable regularizer g, i.e.,

$$g(x) = \sum_{j=1}^{d} g_j(x_j), \tag{6}$$

where $g_j : \mathbb{R} \to \mathbb{R}$ is a univariate function of x_j. Obviously, both the ℓ_1- and ℓ_2-norm regularizers satisfy decomposability.

A favorable choice for solving problem (5) is the primal-dual hybrid gradient (PDHG) algorithm (Chambolle and Pock, 2011). In PDHG, the primal variable x and dual variable y are alternately updated by the proximal gradient method, and the sequence will finally converge to the saddle point of $F(x, y)$. Although PDHG is easy to implement, it is still computationally expensive when the number of examples n is very large. To avoid evaluating full gradients in PDHG whose per-iteration complexity is $O(nd)$, a collection of stochastic primal-dual methods are proposed. Representative algorithms include SPDC

(Zhang and Xiao, 2017), DSPDC (Yu et al., 2015), and SPD1-VR (Tan et al., 2018). SPDC introduces randomness to the selection of dual coordinates, achieving significantly reduced $O(d)$ per-iteration computational cost. DSPDC and SPD1-VR are doubly stochastic algorithms where both the primal and dual coordinates are randomly selected. DSPDC and SPD1-VR have only $O(n + d)$ and $O(1)$ per-iteration complexity respectively, while SPD1-VR requires computation of full gradients once every fixed number of steps.

Although researchers have studied non-stationary sampling probabilities for stochastic gradient methods (Needell et al., 2014; Zhao and Zhang, 2015; Csiba and Richtárik, 2018; Zhou et al., 2018; Horváth and Richtarik, 2019; Qian et al., 2019). and random coordinate methods (Shalev-Shwartz and Tewari, 2011; Nesterov, 2012; Shalev-Shwartz and Zhang, 2013; Richtárik and Takáč, 2014; Lu and Xiao, 2015; Richtárik and Takáč, 2016; Shalev-Shwartz, 2016; Zhang and Gu, 2016) to solve the primal problem (1), there has not been any work that equips stochastic primal-dual methods with adaptive sampling strategies. Taking SPDC as an example, the stationary sampling probability of the dual coordinate y_k is either $1/n$, which uses no extra information, or $(1 - \delta)(1/n) + \delta(\|a_k\|_2 / \sum_{i=1}^{n} \|a_i\|_2)$, which only uses the global property of objective function. Hence, we are motivated to design varying sampling distributions that can exploit more local information of the objective function, which may further improve the convergence performance.

In this paper, we propose an adaptive coordinate sampling strategy, which dynamically adjusts the sampling distribution based on first-order information collected in the past iterations. The sampling probability of each coordinate is weighted by the so-called gradient map, which is very lightweight in terms of both computation and storage. The probability update and adaptive sampling can be efficiently implemented using a binary tree data structure. Specifically, we propose SPDC-AIS, DSPDC-AIS and SPD1-VR-AIS, which are respectively SPDC, DSPDC, and SPD1-VR incorporated with our adaptive sampling strategy. For SPDC-AIS, only dual coordinates are sampled based on some adaptive probabilities, while for DSPDC-AIS, and SPD1-VR-AIS, both the primal and dual coordinates are sampled adaptively. Theoretical analysis of our adaptive coordinate sampling strategy is based on SPDC-AIS. We prove that SPDC-AIS converges linearly for strongly convex objective function. In addition, under proper assumptions that can be empirically verified, we show that the ideal form of our adaptive importance sampling contributes to faster convergence than the vanilla SPDC. Numerical evaluations on widely used support vector machine (SVM) show that notably sharper convergence can be achieved compared with the stochastic primal-dual methods with uniform sampling and traditional non-uniform sampling methods.

2. SINGLY ADAPTIVE SAMPLING

In this section, we introduce our adaptive sampling rule and apply it to SPDC, where only the dual coordinates are randomly chosen.

SPDC is basically an stochastic extension of PDHG (Chambolle and Pock, 2011) to approach the saddle point of $F(x, y)$ of problem (5). They both alternately maximize F with respect to y and minimize

F with respect to x. If we rewrite problem (5) as

$$\min_{x\in\mathbb{R}^d}\max_{y\in\mathbb{R}^n}\left\{F(x,y)=\frac{1}{n}\sum_{i=1}^{n}\left(\langle a_i,x\rangle\, y_i-\phi_i^*(y_i)\right)+g(x)\right\},$$

we can observe that for a fixed x, $F(x,y)$ is decomposable in terms of the dual coordinates y_i's. In the t-th iteration of SPDC, for fixed $\bar{x}^t$, we first uniformly choose an index $i_t\in\{1,\ldots,n\}$ and then perform a proximal gradient ascent step on $\langle a_{i_t},x\rangle\, y_{i_t}-\phi_{i_t}^*(y_{i_t})$. That is,

$$y_{i_t}^{t+1}=\mathrm{prox}_{\sigma\phi_{i_t}^*}(y_{i_t}^t+\sigma\left\langle a_{i_t},\bar{x}^t\right\rangle)=\arg\max_{u\in\mathbb{R}}\left\{\left\langle a_{i_t},\bar{x}^t\right\rangle u-\phi_{i_t}^*(u)-\frac{1}{2\sigma}(u-y_{i_t}^t)^2\right\},\tag{7}$$

where σ is the dual step size. Subsequently, the whole primal vector is updated for fixed y^{t+1}:

$$x^{t+1}=\arg\min_{x\in\mathbb{R}^d}\left\{\left\langle s^t+(y_{i_t}^{t+1}-y_{i_t}^t)a_{i_t},x\right\rangle+g(x)+\frac{1}{2\tau}\|x-x^t\|_2^2\right\},\tag{8}$$

where τ is the primal step size and $s^t=(1/n)\sum_{i=1}^n y_i^t a_i\in\mathbb{R}^d$ can be pre-computed and stored. An extrapolation step is required to facilitate the convergence:

$$\bar{x}^{t+1}=x^{t+1}+\theta(x^{t+1}-x^t).\tag{9}$$

Steps (7)-(9) yield $O(d)$ overall computational cost, which is much lower than the $O(nd)$ per-iteration cost of PDHG.

2.1. SPDC-AIS

To make the algorithm more adaptive to the local structures of $F(x,y)$, we are motivated to design an adaptive coordinate sampling rule for SPDC. Let us rewrite equation (7) as

$$y_{i_t}^{t+1}=y_{i_t}^t+\sigma\mathcal{G}_\sigma(y_{i_t}^t),\tag{10}$$

where

$$\mathcal{G}_\sigma(y_{i_t}^t)=\frac{1}{\sigma}\left[\mathrm{prox}_{\sigma\phi_{i_t}^*}\left(y_{i_t}^t-\sigma\left\langle a_{i_t},x^t\right\rangle\right)-y_{i_t}^t\right]\tag{11}$$

denotes the gradient map. By the first-order optimality condition of proximal mapping, $|\mathcal{G}_\sigma(y_i)|=0$ if and only if y_i maximizes $\langle a_i,x\rangle\, y_i-\phi_i^*(y_i)$. Intuitively speaking, the larger the value of $|\mathcal{G}_\sigma(y_{i_t}^t)|$, the more often we wish to sample the i_t-th dual coordinate. Motivated by this, in the t-th iteration, we can sample the dual index $i\in\{1,\ldots,n\}$ with probability proportional to $|\mathcal{G}_\sigma(y_i^t)|^\kappa$, where κ usually takes non-negative values like 0, 0.5 and 1 (Allen-Zhu et al., 2016) (Nesterov, 2012). However, $|\mathcal{G}_\sigma(y_i^t)|^\kappa$ can be arbitrarily small, leading to little chance of sampling the i-th coordinate. Hence, a sampling distribution that is mixed with the uniform distribution is more reasonable. For instance, the probability

of sampling i-th dual coordinate can be

$$p_i^t = (1-\delta_t)\frac{1}{n} + \delta_t \frac{|\mathcal{G}_\sigma(y_i^t)|^\kappa}{\sum_{k=1}^n |\mathcal{G}_\sigma(y_k^t)|^\kappa},\ \forall i \in \{1,\dots,n\}, \tag{12}$$

where $\delta \in (0,1]$ is the parameter used to balance these two distribution. Hence, p_i^t is lower bounded, i.e., $p_i^t \geq (1-\delta_t)/n$. Nevertheless, (12) is an ideal sampling distribution since it requires the evaluation of $\mathcal{G}_\sigma(y_i^t)$ for all $i \in \{1,\dots,n\}$, which takes $O(nd)$ computational cost at every iteration and is thus not feasible. In our proposed method SPDC-AIS (see Algorithm 1 for details), we overcome this defect by replacing each $\mathcal{G}_\sigma(y_i^t)$ with $\mathcal{G}_\sigma(y_i^{[i]})$, where $[i]$ denotes the most recent iteration in which the index i is picked. We need to store and maintain a vector $\pi = [\pi_1,\dots,\pi_n]$, where $\pi_i = \mathcal{G}_\sigma(y_i^{[i]})$ for $i \in \{1,\dots,n\}$. In other words, we use the historical gradient maps that are evaluated at different iterates to approximate the sampling probabilities (12), in exchange for significantly lower per-iteration computational cost.

Algorithm 1 SPDC-AIS

1: **Input:** primal step size $\tau > 0$, dual step size $\sigma > 0$, number of iterations T, initial points x^0 and y^0, parameters $\delta_t \in [\underline{\delta}, \bar{\delta}]$, $\kappa, \theta > 0$.
2: **Initialize:** $\bar{x}^0 = x^0$, $s^0 = (1/n)\sum_{k=1}^n y_k^0 a_k$, $\pi_i = 1$ for all $i \in \{1,\dots,n\}$
3: **for** $t = 0, 1, 2, \dots, T-1$ **do**
4: Update probability distribution p^t, where

$$p_i^t = (1-\delta_t)\frac{1}{n} + \delta_t \frac{|\pi_i|^\kappa}{\sum_{k=1}^n |\pi_k|^\kappa},\ \forall i \in \{1,\dots,n\} \tag{13}$$

5: Randomly pick $i_t \in \{1,2,\dots,n\}$ according to the distribution p^t
6: Perform updates:

$$\begin{aligned}
y_{i_t}^{t+1} &= \arg\max_{\beta\in\mathbb{R}} \left\{ \left\langle a_i, \bar{x}^t \right\rangle \beta - \phi_i^*(\beta) - \frac{np_{i_t}^t}{2\sigma}(\beta - y_i^t)^2 \right\} \\
y_i^{t+1} &= y_i^t \text{ for all } i \neq i_t \\
\pi_{i_t} &= \frac{np_{i_t}^t}{\sigma}\left(y_{i_t}^{t+1} - y_{i_t}^t\right) \\
x^{t+1} &= \arg\min_{x\in\mathbb{R}^d} \left\{ \left\langle s^t + \frac{y_{i_t}^{t+1} - y_{i_t}^t}{np_{i_t}^t} a_{i_t}, x \right\rangle + g(x) + \frac{\|x - x^t\|_2^2}{2\tau} \right\} \\
s^{t+1} &= s^t + \frac{1}{n}(y_{i_t}^{t+1} - y_{i_t}^t)a_{i_t} \\
\bar{x}^{t+1} &= x^{t+1} + \theta(x^{t+1} - x^t)
\end{aligned}$$

7: **end for**
8: **Output:** x^T and y^T

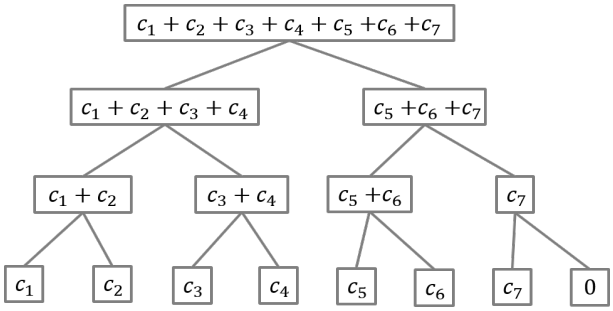


Fig. 1: Example of Binary Tree $\mathcal{T}_7$ (Each $c_i = |\pi_i|^\kappa$)

2.2. Implementation of the Sampling

For SPDC-AIS, we need to choose the indices according to a varying non-uniform distribution. In other words, we need to generate a random integer that follows a different distribution p in every iteration. To achieve this in a computationally efficient way, we resort to a binary tree data structure.

Suppose that there are n data examples. Let $c_i = |\pi_i|^\kappa$ be the i-th leaf of $\mathcal{T}_n$. By properly adding "empty" nodes, we can group the nodes at each level in pairs. Fig. 1 illustrates an example of such a binary tree with $n = 7$. Note that we need to add one more leaf at the bottom level, so that it can be grouped with c_7. According to Algorithm 1, some π_i is changed in each iteration. Hence, all nodes related to c_i in $\mathcal{T}_n$ should be updated. The update of $\mathcal{T}_n$ can be done in a bottom-up way. It is known that the height of $\mathcal{T}_n$ is $\lceil \log n \rceil$. Thus, by updating the tree in a bottom-up approach, the computational cost is $O(\log n)$. $\mathcal{T}_n$ can contribute to generating a random integer following a given distribution. If we first generate a uniformly distributed random number $r \in [0, 1]$, we are supposed to find the index i such that $\sum_{k=1}^{i-1} p_k^t < r < \sum_{k=1}^{i} p_k^t$, where for $i \in \{1, \ldots, n\}$,

$$\sum_{k=1}^{i} p_k^t = \frac{(1-\delta_t)i}{n} + \delta_t \frac{\sum_{k=1}^{i} |\pi_k|^\kappa}{\sum_{l=1}^{n} |\pi_l|^\kappa}. \tag{14}$$

Since all the partial sums on the right-hand side of (14) are stored in $\mathcal{T}_n$, we can quickly find the desired i by visiting $\mathcal{T}_n$ in a top-down fashion. Obviously, searching for i also takes $O(\log n)$ time.

To conclude, additional $O(\log n)$ per-iteration cost is needed to implement the adaptive sampling. Since updating the primal and dual variables already requires $O(d)$ per-iteration cost, our adaptive coordinate sampling method does not increase the order of computational complexity.

2.3. Convergence Analysis

In this section, we provide a theoretical analysis of the proposed adaptive coordinate sampling method SPDC-AIS.

SPDC-AIS shown in Algorithm 1 requires three control parameters τ, σ, and θ, and its convergence

is guaranteed provided that the parameters are properly specified and the sampling probabilities do not vary drastically. Specifically, we have the following Theorem.

Theorem 1. *Let $\{x^t, y^t\}$ be the sequence generated by Algorithm 1. Suppose that each ϕ_i is convex and $(1/\gamma)$-smooth, g is λ-strongly convex. Denote $R := \max_i \|a_i\|_2$. The parameters τ, σ, θ in Algorithm 1 are chosen as*

$$\tau = \frac{1-\overline{\delta}}{2R}\sqrt{\frac{\gamma}{n\lambda}}, \quad \sigma = \frac{1-\overline{\delta}}{2R}\sqrt{\frac{n\lambda}{\gamma}}, \quad \theta = 1-\mu,$$

where $\mu = \min\left\{\frac{2\lambda\tau}{1+2\lambda\tau}, \frac{\gamma}{n/\sigma + n/(1-\overline{\delta})}\right\}$. Suppose that the following inequality holds for all $t \geq 1$:

$$\sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma(1-p_i^t)}{np_i^t}\right)(y_i^t - y_i^*)^2 \leq \theta \cdot \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma}{np_i^{t-1}}\right)(y_i^t - y_i^*)^2. \tag{15}$$

Then, we have

$$\begin{aligned}&\mathbb{E}\left[\left(\frac{1}{2\tau}+\lambda\right)\|x^t - x^*\|_2^2 + \sum_{i=1}^{n}\left(\frac{1}{4\sigma}+\frac{\gamma}{n}\right)(y_i^t - y_i^*)^2\right] \\ &\leq \theta^t\left[\left(\frac{1}{2\tau}+\lambda\right)\|x^0 - x^*\|_2^2 + \sum_{i=1}^{n}\left(\frac{1}{2\sigma}+\gamma\right)(y_i^0 - y_i^*)^2\right],\end{aligned}$$

where (x^*, y^*) is the saddle point.

Theorem 1 shows that the sequence $\{(x^t, y^t)\}$ will converge linearly to the unique saddle point in expectation, which matches the results in (Zhang and Xiao, 2017). Before we present the detailed proof of Theorem 1, we conduct simulations on three datasets (colon-cancer, a2a, gisette) to check the assumption (15). We obtain an almost optimal value y^* by running SPDC for sufficiently large number of iterations, then we compute the term

$$\theta_t = \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma(1-p_i^t)}{np_i^t}\right)(y_i^t - y_i^*)^2 \Bigg/ \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma}{np_i^{t-1}}\right)(y_i^t - y_i^*)^2,$$

and compare θ_t with θ. The results are presented in Fig. 2, which shows that $\theta^t \leq \theta$ holds in most of the iterations. Therefore, we claim that the condition (15) holds on average empirically.

To prove Theorem 1, we first state the following key lemma, which is a direct consequence of inequalities (62) and (63) in (Zhang and Xiao, 2017).

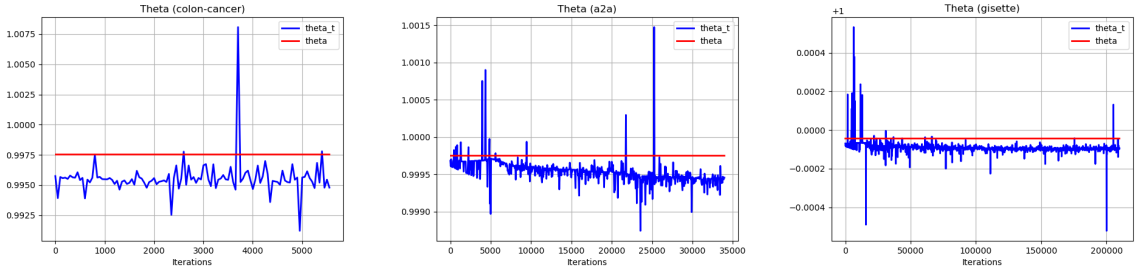


Fig. 2: Comparison of θ_t and θ for SPD1-AIS Algorithms

Lemma 1. *Let $\{(x^t, y^t)\}$ be the sequence generated by Algorithm 1. Then, we have*

$$\begin{aligned}
&\frac{\|x^t - x^*\|_2^2}{2\tau} + \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma(1-p_i^t)}{np_i^t}\right)(y_i^t - y_i^*)^2 \geq \\
&\mathbb{E}\Bigg[\left(\frac{1}{2\tau} + \lambda\right)\|x^{t+1} - x^*\|_2^2 + \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma}{np_{i_t}^t}\right)(y_{i_t}^{t+1} - y_{i_t}^*)^2 \\
&\quad + \frac{(y^{t+1} - y^*)^T A(x^{t+1} - x^t)}{n} - \frac{\theta(y^t - y^*)^T A(x^t - x^{t-1})}{n} \\
&\quad + \frac{\|x^{t+1} - x^t\|_2^2}{2\tau} + \frac{(y_{i_t}^{t+1} - y_{i_t}^t)^2}{2\sigma} \\
&\quad - \frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1} - y_{i_t}^t\|_2\|a_{i_t}\|_2\left(\|x^{t+1} - x^t\|_2 + \theta\|x^t - x^{t-1}\|_2\right) \mid \mathcal{F}_t\Bigg].
\end{aligned} \tag{16}$$

Based on Lemma 1, we can obtain the following proposition.

Proposition 1. *Assume σ, τ are chosen such that $\sigma\tau \leq (1-\overline{\delta})^2/4R^2$. Then, we have*

$$\begin{aligned}
&\frac{\|x^t - x^*\|_2^2}{2\tau} + \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma(1-p_i^t)}{np_i^t}\right)(y_i^t - y_i^*)^2 + \frac{\theta(y^t - y^*)^T A(x^t - x^{t-1})}{n} + \frac{\theta\|x^t - x^{t-1}\|_2^2}{4\tau} \\
&\geq \mathbb{E}\Bigg[(\frac{1}{2\tau} + \lambda)\|x^{t+1} - x^*\|_2^2 + \sum_{i=1}^{n}\left(\frac{1}{2\sigma} + \frac{\gamma}{np_i^t}\right)(y_i^{t+1} - y_i^*)^2 \\
&\qquad + \frac{(y^{t+1} - y^*)^T A(x^{t+1} - x^t)}{n} + \frac{\|x^{t+1} - x^t\|_2^2}{4\tau} \mid \mathcal{F}_t\Bigg].
\end{aligned}$$

Proof. We need to lower bound the last term on the right-hand side of inequality (16). Observe that we

have

$$\begin{aligned}\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2\|a_{i_t}\|_2\|x^{t+1}-x^t\|_2 &\le \frac{\|x^{t+1}-x^t\|_2^2}{4\tau}+\frac{\tau}{(np_{i_t}^t)^2}\|a_{i_t}\|_2^2\|y_{i_t}^{t+1}-y_{i_t}^t\|_2^2\\ &\le \frac{\|x^{t+1}-x^t\|_2^2}{4\tau}+\frac{\tau R^2}{(1-\overline{\delta})^2}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2^2\\ &\le \frac{\|x^{t+1}-x^t\|_2^2}{4\tau}+\frac{1}{4\sigma}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2^2.\end{aligned}$$

The first inequality holds because of Young's inequality; the second holds since we know that $np_{i_t}^t \ge 1-\delta_t \ge 1-\overline{\delta}$ and $\|a_{i_t}\|_2 \le R$; the last inequality holds because of the assumption that $\tau \le (1-\overline{\delta})^2/4R^2\sigma$. Similarly, we have

$$\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1}-y_t^{i_t}\|_2\|a_{i_t}\|_2\|x^t-x^{t-1}\|_2 \le \frac{\|x^t-x^{t-1}\|_2^2}{4\tau}+\frac{1}{4\sigma}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2^2.$$

Thus, the last term on the right-hand side of inequality (16) can be lower bounded by

$$\begin{aligned}&\mathbb{E}\Bigg[\frac{\|x^{t+1}-x^t\|_2^2}{2\tau}+\frac{(y_{i_t}^{t+1}-y_{i_t}^t)^2}{2\sigma}\\ &\qquad -\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2\|a_{i_t}\|_2\left(\|x^{t+1}-x^t\|_2+\theta\|x^t-x^{t-1}\|_2\right)\mid \mathcal{F}_t\Bigg]\\ \le\ &\mathbb{E}\left[\frac{\|x^{t+1}-x^t\|_2^2}{4\tau}-\theta\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}\right].\end{aligned}$$

Combining the above inequality and (16) completes the proof. □

Now, we are ready to prove the Theorem 1.

Proof of Theorem 1. Define Δ^t ($t \ge 0$) as

$$\begin{aligned}\Delta^t = \mathbb{E}\Bigg[\left(\frac{1}{2\tau}+\lambda\right)\|x^t-x^*\|_2^2+\sum_{i=1}^n\left(\frac{1}{2\sigma}+\frac{\gamma}{np_i^{t-1}}\right)(y_i^t-y_i^*)^2\\ +\frac{(y^t-y^*)^TA(x^t-x^{t-1})}{n}+\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}\Bigg].\end{aligned}$$

First, based on the assignment of the parameters, we have

$$\frac{1/(2\tau)}{1/(2\tau)+\lambda} \le \theta.$$

Then, combining (15) and Proposition 1, we obtain the recursive relation $\Delta^{t+1} \leq \theta \cdot \Delta^t$. Thus,

$$\mathbb{E}\Bigg[\left(\frac{1}{2\tau}+\lambda\right)\|x^t-x^*\|_2^2+\sum_{i=1}^n\left(\frac{1}{2\sigma}+\frac{\gamma}{np_i^{t-1}}\right)(y_i^t-y_i^*)^2 \\ +\frac{(y^t-y^*)^TA(x^t-x^{t-1})}{n}+\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}\Bigg]\leq\theta^t\Delta^0, \tag{17}$$

where

$$\Delta^0=\left(\frac{1}{2\tau}+\lambda\right)\|x^0-x^*\|_2^2+\sum_{i=1}^n\left(\frac{1}{2\sigma}+\gamma\right)(y_i^0-y_i^*)^2.$$

To bound the last two terms on the left-hand side of inequality (17), we note that

$$\begin{aligned}&\frac{(y^t-y^*)^TA(x^t-x^{t-1})}{n}\\ \geq&-\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}-\frac{\tau\|y^t-y^*\|_2^2\|A\|_2^2}{n^2}\\ \geq&-\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}-\frac{(1-\overline{\delta})^2\|y^t-y^*\|_2^2}{4n\sigma}\\ \geq&-\frac{\|x^t-x^{t-1}\|_2^2}{4\tau}-\frac{\|y^t-y^*\|_2^2}{4\sigma}.\end{aligned}$$

The first inequality holds because of Young's inequality and the second inequality follows from the assumption $\tau \leq (1-\overline{\delta})^2/4R^2\sigma$. Finally, we can simplify inequality (17) as

$$\mathbb{E}\left[\left(\frac{1}{2\tau}+\lambda\right)\|x^t-x^*\|_2^2+\sum_{i=1}^n\left(\frac{1}{4\sigma}+\frac{\gamma}{np_i^{t-1}}\right)(y_i^t-y_i^*)^2\right]\leq\theta^t\Delta^0. \tag{18}$$

Applying the fact that $p_i^{t-1} \leq 1$ to (18), the proof is completed. □

Theorem 1 demonstrates that SPDC-AIS has a comparable convergence rate with SPDC, while theoretically it does not demonstrate the advantages of importance sampling. In the following theorem, we show that importance sampling does enjoy a faster convergence rate. Before that, we need the following assumption.

Assumption 1. *There exists a constant $\rho > 0$ such that for any $y \in \mathbb{R}^n$ and $\sigma > 0$, we have*

$$\frac{\sum_{i=1}^n|\mathcal{G}_\sigma(y_i)|^3}{\sum_{i=1}^n|\mathcal{G}_\sigma(y_i)|}-\left(\frac{1}{n}\sum_{i=1}^n|\mathcal{G}_\sigma(y_i)|\right)^2\geq\frac{\rho}{n}\|y-y^*\|_2^2, \tag{19}$$

where $\mathcal{G}_\sigma(y_i)$ is the proximal mapping defined in (11).

According to the generalized mean inequality, we have

$$\frac{1}{n}\sum_{i=1}^{n}|\mathcal{G}_\sigma(y_i)|^3 \geq \left(\frac{1}{n}\sum_{i=1}^{n}|\mathcal{G}_\sigma(y_i)|\right)^3.$$

Thus, the left-hand-side of (19) is always non-negative and the equality holds only when all $\mathcal{G}_\sigma(y_i)$'s take the same value. Besides, we conduct simulations to approximate the exact value of ρ for three different datasets (colon-cancer, a2a, gisette). We first obtain an almost optimal value y^* by running SPDC for a sufficient large number of iterations, then we compute the following term in each iteration:

$$\rho_t = \left(\frac{\sum_{i=1}^{n}|\mathcal{G}_\sigma(y_i^t)|^3}{\sum_{i=1}^{n}|\mathcal{G}_\sigma(y_i^t)|} - \left(\frac{1}{n}\sum_{i=1}^{n}|\mathcal{G}_\sigma(y_i^t)|\right)^2\right) \Bigg/ \left(\frac{1}{n}\|y^t - y^*\|_2^2\right).$$

The results are shown in Fig. 3. Then, we choose the minimum value of ρ_t as a reasonable approximation

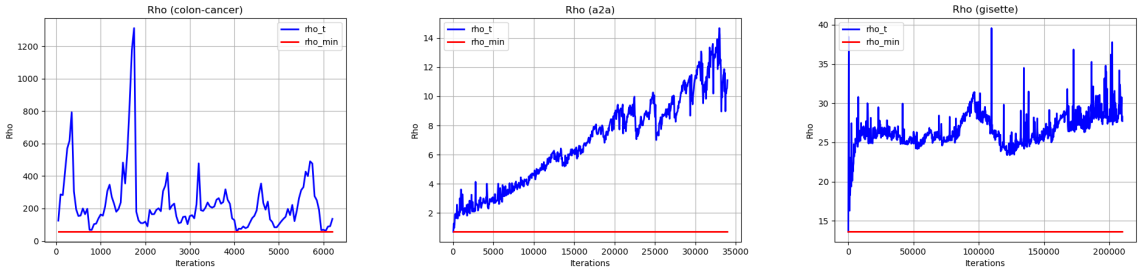


Fig. 3: The Value of ρ_t in Each Iteration of SPD1-AIS

of ρ in Assumption 1. Table 1 reports the specific choices of ρ for the three different datasets. As it is observed from Fig. 3 and Table 1, Assumption 1 always holds with some large enough constant $\rho > 0$, especially when y is close to y^*.

Table 1: Parameters for Different Datasets

	colon-cancer	a2a	gisette
ρ	56.6	0.7	13.6

Equipped with Assumption 1, we can establish an improved rate of convergence by adopting the adaptive probability distribution given in (12).

Theorem 2. *Suppose that each ϕ_i is convex, $(1/\gamma)$-smooth, g is λ-strongly convex and Assumption 1 holds. Besides, assume that*

$$p_i^t = (1-\delta_t)\frac{1}{n} + \delta_t\frac{|\mathcal{G}_\sigma(y_i^t)|}{\sum_{k=1}^{n}|\mathcal{G}_\sigma(y_k^t)|},\ \forall i \in \{1,\ldots,n\}.$$

Denote $R := \max_i \|a_i\|_2$. *If the parameters* τ, σ, θ *are chosen as*

$$\tau = \frac{1}{2R}\sqrt{\frac{\gamma}{n\lambda}}, \quad \sigma = \frac{1}{2R}\sqrt{\frac{n\lambda}{\gamma}}, \quad \theta = 1 - \tilde{\mu},$$

where $\tilde{\mu} = \min\left\{\frac{2\lambda\tau}{1+2\lambda\tau}, \frac{\gamma+\rho\sigma\underline{\delta}}{n/\sigma+n/(1-\overline{\delta})}\right\}$, *then we have*

$$\begin{aligned}
&\mathbb{E}\left[\left(\frac{1}{2\tau}+\lambda\right)\|x^t - x^*\|_2^2 + \sum_{i=1}^n \left(\frac{1}{4\sigma}+\frac{\gamma}{n}\right)(y_i^t - y_i^*)^2\right] \\
&\leq \theta^t\left[\left(\frac{1}{2\tau}+\lambda\right)\|x^0 - x^*\|_2^2 + \sum_{i=1}^n \left(\frac{1}{2\sigma}+\gamma\right)(y_i^0 - y_i^*)^2\right],
\end{aligned}$$

where (x^*, y^*) *is the saddle point.*

Proof. We know that

$$\mathbb{E}\left[\frac{(y_{i_t}^{t+1} - y_{i_t}^t)^2}{2\sigma} \mid \mathcal{F}_t\right] = \sum_{i=1}^n \frac{\sigma}{2}|\mathcal{G}_\sigma(y_i^t)|^2 p_i^t = \frac{(1-\delta_t)\sigma}{2n}\sum_{i=1}^n |\mathcal{G}_\sigma(y_i^t)|^2 + \frac{\delta_t\sigma}{2}\frac{\sum_{i=1}^n |\mathcal{G}_\sigma(y_i^t)|^3}{\sum_{i=1}^n |\mathcal{G}_\sigma(y_i^t)|}. \tag{20}$$

By the definition of p_i^t and the fact that $(ax+by)(a/x+b/y) \geq (a+b)^2$ for all x, y, a, $b > 0$, we have

$$\frac{1}{p_i^t} \leq (1-\delta_t)n + \delta_t\frac{\sum_{k=1}^n |\mathcal{G}_\sigma(y_k^t)|}{|\mathcal{G}_\sigma(y_i^t)|}. \tag{21}$$

Thus, we lower bound the last term on the right-hand side of inequality (16)

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1} - y_{i_t}^t\|_2\|a_{i_t}\|_2\|x^{t+1} - x^t\|_2 \mid \mathcal{F}_t\right] \\
&\leq \mathbb{E}\left[\frac{\|x^{t+1} - x^t\|_2^2}{4\tau} + \frac{\tau}{(np_{i_t}^t)^2}\|a_{i_t}\|_2^2\|y_{i_t}^{t+1} - y_{i_t}^t\|_2^2 \mid \mathcal{F}_t\right] \\
&= \mathbb{E}\left[\frac{\|x^{t+1} - x^t\|_2^2}{4\tau} \mid \mathcal{F}_t\right] + \sum_{i=1}^n \frac{\tau R^2}{n^2 p_i^t}\sigma^2|\mathcal{G}_\sigma(y_i^t)|^2 \\
&\leq \mathbb{E}\left[\frac{\|x^{t+1} - x^t\|_2^2}{4\tau} \mid \mathcal{F}_t\right] + \frac{(1-\delta_t)\sigma}{4n}\sum_{i=1}^n |\mathcal{G}_\sigma(y_i^t)|^2 + \frac{\delta_t\sigma}{4n^2}\left(\sum_{i=1}^n |\mathcal{G}_\sigma(y_i^t)|\right)^2,
\end{aligned} \tag{22}$$

where the last inequality holds because of (21) and the fact that $\tau\sigma = 1/4R^2$. Similarly with (22), we

also have

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2\|a_{i_t}\|_2\|x^t-x^{t-1}\|_2 \mid \mathcal{F}_t\right] \\
&\leq \mathbb{E}\left[\frac{\|x^t-x^{t-1}\|_2^2}{4\tau} \mid \mathcal{F}_t\right]+\frac{(1-\delta_t)\sigma}{4n}\sum_{i=1}^n|\mathcal{G}_\sigma(y_i^t)|^2+\frac{\delta_t\sigma}{4n^2}\left(\sum_{i=1}^n|\mathcal{G}_\sigma(y_i^t)|\right)^2.
\end{aligned} \tag{23}$$

Combining (20), (22), and (23) gives

$$\begin{aligned}
&\mathbb{E}\Bigg[\frac{\|x^{t+1}-x^t\|_2^2}{2\tau}+\frac{(y_{i_t}^{t+1}-y_{i_t}^t)^2}{2\sigma} \\
&\qquad -\frac{1}{np_{i_t}^t}\|y_{i_t}^{t+1}-y_{i_t}^t\|_2\|a_{i_t}\|_2(\|x^{t+1}-x^t\|_2+\theta\|x^t-x^{t-1}\|_2) \mid \mathcal{F}_t\Bigg] \\
&\geq \mathbb{E}\left[\frac{\|x^{t+1}-x^t\|_2^2}{4\tau}-\theta\frac{\|x^t-x^{t-1}\|_2^2}{4\tau} \mid \mathcal{F}_t\right]+\frac{\delta_t\sigma}{2}\left(\frac{\sum_{i=1}^n|\mathcal{G}_\sigma(y_i^t)|^3}{\sum_{i=1}^n|\mathcal{G}_\sigma(y_i^t)|}-(\frac{1}{n}\sum_{i=1}^n|\mathcal{G}_\sigma(y_i^t)|)^2\right) \\
&\geq \mathbb{E}\left[\frac{\|x^{t+1}-x^t\|_2^2}{4\tau}-\theta\frac{\|x^t-x^{t-1}\|_2^2}{4\tau} \mid \mathcal{F}_t\right]+\frac{\delta_t\sigma\rho}{2n}\|y^t-y^*\|_2^2,
\end{aligned} \tag{24}$$

where the last inequality follows from Assumption 1. Furthermore, by Lemma 1, we obtain an improved version of Proposition 1:

$$\begin{aligned}
&\frac{\|x^t-x^*\|_2^2}{2\tau}+\sum_{i=1}^n\left(\frac{1}{2\sigma}-\frac{\delta_t\sigma\rho}{2}+\frac{\gamma(1-p_i^t)}{np_i^t}\right)(y_i^t-y_i^*)^2 \\
&\qquad +\frac{\theta(y^t-y^*)^TA(x^t-x^{t-1})}{n}+\theta\frac{\|x^t-x^{t-1}\|_2^2}{4\tau} \\
&\geq \mathbb{E}\Bigg[(\frac{1}{2\tau}+\lambda)\|x^{t+1}-x^*\|_2^2+\sum_{i=1}^n\left(\frac{1}{2\sigma}+\frac{\gamma}{np_i^t}\right)(y_i^{t+1}-y_i^*)^2 \\
&\qquad +\frac{(y^{t+1}-y^*)^TA(x^{t+1}-x^t)}{n}+\frac{\|x^{t+1}-x^t\|_2^2}{4\tau} \mid \mathcal{F}_t\Bigg].
\end{aligned}$$

The remaining part of the proof is just the same as the proof of Theorem 1. This completes the proof of Theorem 2. □

Comparing Theorem 2 with Theorem 1, we can see that $\tilde{\mu} > \mu$, which actually indicates the benefit brought by the importance sampling. In other words, under proper assumptions, the theoretical convergence rate is improved from $1-\mu$ to $1-\tilde{\mu}$.

Remark 1. According to Theorem 1 in (Zhang and Xiao, 2017), SPDC with the same parameters τ and

σ achieve the following convergence rate:

$$\mathbb{E}[\Delta^{(t)}] \leq \overline{\theta}^t \left(\Delta^{(0)} + \frac{\|y^{(0)} - y^\star\|_2^2}{4\sigma} \right) \quad \text{where} \quad \overline{\theta} = 1 - \left(n + 2R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1}.$$

To compare the convergence rates of SPDC and SPDC-AIS, we only need to compare $\overline{\theta}$ and θ given in Theorem 2. Note that

$$\begin{aligned}
\theta = 1 - \tilde{\mu} &= 1 - \min\left\{ \frac{2\lambda\tau}{1+2\lambda\tau}, \frac{\gamma + \rho\sigma\underline{\delta}}{n/\sigma + n/(1-\overline{\delta})} \right\} \\
&= \max\left\{ 1 - \left(1 + R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1}, 1 - \left(\frac{n}{(\gamma+\rho\sigma\underline{\delta})(1-\overline{\delta})} + \frac{\gamma}{\gamma+\rho\sigma\underline{\delta}} \cdot 2R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1} \right\}.
\end{aligned}$$

Assuming $\gamma \geq 1$ (which is true for the three loss functions mentioned in Introduction), by choosing $\overline{\delta} = 1 - 1/(\gamma + \rho\sigma\underline{\delta}) \in (0,1)$, we have

$$\begin{aligned}
\theta &= \max\left\{ 1 - \left(1 + R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1}, 1 - \left(n + \frac{\gamma}{\gamma+\rho\sigma\underline{\delta}} \cdot 2R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1} \right\} \\
&\leq 1 - \left(n + \max\left\{ \frac{\gamma}{\gamma+\rho\sigma\underline{\delta}}, \frac{1}{2} \right\} \cdot 2R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1} \\
&< 1 - \left(n + 2R\sqrt{\frac{n}{\lambda\gamma}} \right)^{-1} \\
&= \overline{\theta},
\end{aligned}$$

where the first inequality is due to $\max\left\{ \frac{\gamma}{\gamma+\rho\sigma\underline{\delta}}, \frac{1}{2} \right\} < 1$. Thus, ρ depends on how much the convergence rate is improved. Besides, if there is no importance sampling, i.e., $\underline{\delta} = \overline{\delta} = 0$, then we have $\theta = \overline{\theta}$, which means that SPDC-AIS reduces to the standard SPDC.

3. DOUBLY ADAPTIVE SAMPLING

In this section, we extend the adaptive coordinate sampling strategy to the doubly stochastic algorithms DSPDC (Yu et al., 2015) and SPD1-VR (Tan et al., 2018), where the coordinates of both the primal and dual variables are randomly sampled.

Algorithm 2 DSPDC-AIS

1: **Input:** primal step size $\tau > 0$, dual step size $\sigma > 0$, number of iterations T, initial points x^0 and y^0, parameters $\delta_t \in [\underline{\delta}, \bar{\delta}]$, $\kappa, \theta > 0$.
2: **Initialize:** $\bar{x}^0 = x^0$, $s^0 = (1/n)\sum_{k=1}^n y_k^0 a_k$, $\pi_i = \psi_j = 1$ for all $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, d\}$
3: **for** $t = 0, 1, 2, \ldots, T-1$ **do**
4: Update probability distribution p^t and q^t, where

$$p_i^t = (1 - \delta_t)\frac{1}{n} + \delta_t \frac{|\pi_i|^\kappa}{\sum_{k=1}^n |\pi_k|^\kappa}, \ \forall i \in \{1, \ldots, n\} \tag{25}$$

$$q_j^t = (1 - \delta_t)\frac{1}{d} + \delta_t \frac{|\psi_j|^\kappa}{\sum_{k=1}^d |\psi_k|^\kappa}, \ \forall j \in \{1, \ldots, d\} \tag{26}$$

5: Randomly pick $i_t \in \{1, 2, \ldots, n\}$ and $j_t \in \{1, 2, \ldots, d\}$ according to the distribution p^t and q^t, respectively
6: Perform updates:

$$\begin{aligned}
y_{i_t}^{t+1} &= \arg\max_{\beta \in \mathbb{R}} \left\{ \left\langle a_i, \bar{x}^t \right\rangle \beta - \phi_i^*(\beta) - \frac{np_{i_t}^t}{2\sigma}(\beta - y_{i_t}^t)^2 \right\} \\
y_i^{t+1} &= y_i^t \text{ for all } i \neq i_t \\
\pi_{i_t} &= \frac{np_{i_t}^t}{2\sigma}(y_{i_t}^{t+1} - y_{i_t}^t) \\
\bar{y}^{t+1} &= y^{t+1} + n(y^{t+1} - y^t) \\
x_{j_t}^{t+1} &= \arg\max_{\alpha \in \mathbb{R}} \left\{ \frac{1}{n} \left\langle A^j, \bar{y}^{t+1} \right\rangle \alpha - g_j(\alpha) - \frac{dq_{j_t}^t}{2\tau}(\alpha - x_j^t)^2 \right\} \\
x_j^{t+1} &= x_j^t \text{ for all } j \neq j_t \\
\psi_{j_t} &= \frac{dq_{j_t}^t}{2\tau}(x_{j_t}^{t+1} - x_{j_t}^t) \\
\bar{x}^{t+1} &= x^{t+1} + \theta(x^{t+1} - x^t)
\end{aligned}$$

7: **end for**
8: **Output:** x^T and y^T

3.1. DSPDC-AIS

For DSPDC, instead of updating the whole vector x^{t+1}, we randomly sample $j_t \in \{1, 2, \ldots, d\}$ and update $x_{j_t}^{t+1}$ as

$$x_{j_t}^{t+1} = \arg\min_{\alpha \in \mathbb{R}} \left\{ \frac{1}{n} \left\langle A^{j_t}, \bar{y}^{t+1} \right\rangle \alpha + g_{j_t}(\alpha) + \frac{1}{2\tau}(\alpha - x_{j_t}^t)^2 \right\},$$

where A^j denotes the j-th column of A and τ is the primal step size. The importance sampling strategy for the primal variable x is similar to that for the dual variable y. First, we define the gradient mapping for primal variables as

$$\mathcal{G}_\tau(x_j^t) = \left[x_j^t - \text{prox}_{\tau g_j}\left(x_j^t - \frac{\tau}{n}\left\langle A^j, y^t\right\rangle\right)\right]/\tau.$$

Then, the sampling distribution is mixed with the uniform distribution, which is given by

$$q_j^t = (1-\delta_t)\frac{1}{d} + \delta_t \frac{|\mathcal{G}_\tau(x_j^t)|^\kappa}{\sum_{k=1}^d |\mathcal{G}_\tau(x_k^t)|^\kappa},\ \forall j \in \{1,\ldots,d\}. \tag{27}$$

For simplicity, we use the historical gradient maps $\mathcal{G}_\tau(x_i^{[i]})$ evaluated in the most recent iteration to approximate $\mathcal{G}_\tau(x_i^t)$. Besides, we need to use another binary tree, whose i-th leaf is $|\psi_i| = |\mathcal{G}_\tau(x_i^{[i]})|$, to store the gradient information and achieve non-uniform sampling. See Algorithm 2 for details.

3.2. SPD1-VR-AIS

SPD1-VR is a variant of DSPDC. Utilizing the decomposable structure of g as shown in (6), we can further rewrite problem (5) as

$$\min_{x\in\mathbb{R}^d}\max_{y\in\mathbb{R}^n}\left\{\frac{1}{n}\sum_{i=1}^n\sum_{j=1}^d\left(a_{ij}y_ix_j - \frac{1}{d}\phi_i^*(y_i) + g_j(x_j)\right)\right\}.$$

Observe that both the primal and dual coordinates are decomposable by fixing the other one. We can randomly choose a primal index and a dual index in every iteration and achieve only $\mathcal{O}(1)$ per-iteration cost. Besides, variance reduction technique (Johnson and Zhang, 2013) is employed in SPD1-VR to accelerate the convergence, where full gradients with regard to both the primal and dual variables are re-computed once every fixed number of iterations.

Instead of picking the primal and dual coordinates uniformly, we propose SPD1-VR-AIS that incorporates SPD1-VR with our adaptive coordinate sampling strategy (see Algorithm 3). This is a double-loop algorithm and full gradients are computed periodically, which takes $O(nd)$ time every epoch. Instead of updating the probability distribution in every inner iteration, we do it right after computing the full gradients at the beginning of each epoch. The sampling probabilities are also defined by gradient maps as in (12) and (27). The computation of the proximal mappings takes only an additional $O(n+d)$ time for each epoch. Since the epoch length is T, the sampling probability will be calibrated every T iterations. In other words, the sampling in the inner iterations uses historical information no more than T iterations ago. Moreover, non-uniform sampling is also performed at the beginning of each epoch, which guarantees that the per-iteration cost (including sampling and iteration) is still $O(1)$.

Algorithm 3 SPD1-VR-AIS

Input: primal step size $\tau > 0$, dual step size $\sigma > 0$, number of iterations T, initial points x^0 and y^0, parameters $\delta_t \in [\underline{\delta}, \bar{\delta}]$.
Initialize: $\tilde{x}^0 \in X$ and $\tilde{y}^0 \in Y$
for $k = 0, 1, 2, \ldots, K-1$ **do**
  Compute full gradients $G_x^k = (1/n)A^\top \tilde{y}^k$ and $G_y^k = (1/d)A\tilde{x}^k$
  Compute probability distribution p^k and q^k, where

$$p_i^k = (1-\delta_k)\frac{1}{n} + \delta_k \frac{|\mathcal{G}_\sigma(\tilde{y}_i^k)|^\kappa}{\sum_{l=1}^n |\mathcal{G}_\sigma(\tilde{y}_l^k)|^\kappa},\ \forall i \in \{1, \ldots, n\}$$

$$q_j^k = (1-\delta_k)\frac{1}{d} + \delta_k \frac{|\mathcal{G}_\tau(\tilde{x}_j^k)|^\kappa}{\sum_{l=1}^d |\mathcal{G}_\tau(\tilde{x}_l^k)|^\kappa},\ \forall j \in \{1, \ldots, d\}$$

  Independently sample $nd/\log(nd)$ primal indices following p^k, and $nd/\log(nd)$ dual indices following q^k, and store them in set I and J respectively
  Set $(x^0, y^0) = (\tilde{x}^0, \tilde{y}^0)$
  for $t = 0, 1, \ldots, T-1$ **do**
    Uniformly pick $i_t, i_t' \in I$ and $j_t, j_t' \in J$
    Perform updates:

$$\begin{aligned}
\bar{x}_{j_t}^t &= \text{prox}_{\tau g_{j_t}}\left(x_{j_t}^t - \tau\left(a_{i_t' j_t}(y_{i_t'}^t - \tilde{y}_{i_t'}^k) + G_{x,j_t}^k\right)\right)\\
\bar{y}_{i_t}^t &= \text{prox}_{(\sigma/d)\phi_{i_t}^*}\left(y_{i_t}^t - \sigma\left(a_{i_t j_t'}(x_{j_t'}^t - \tilde{x}_{j_t'}^k) + G_{y,i_t}^k\right)\right)\\
x_{j_t}^{t+1} &= \text{prox}_{\tau g_{j_t}}\left(x_{j_t}^t - \tau\left(a_{i_t j_t}(\bar{y}_{i_t}^t - \tilde{y}_{i_t}^k) + G_{x,j_t}^k\right)\right)\\
y_{i_t}^{t+1} &= \text{prox}_{(\sigma/d)\phi_{i_t}^*}\left(y_{i_t}^t - \sigma\left(a_{i_t j_t}(\bar{x}_{j_t}^t - \tilde{x}_{j_t}^k) + G_{y,i_t}^k\right)\right)\\
x_j^{t+1} &= x_j^t \text{ for all } j \neq j_t\\
y_i^{t+1} &= y_i^t \text{ for all } i \neq i_t
\end{aligned}$$

  end for
  Set $(\tilde{x}^{k+1}, \tilde{y}^{k+1}) = (x^T, y^T)$
end for
Output: $\tilde{x}^{k+1}$ and $\tilde{y}^{k+1}$

4. EXPERIMENT RESULTS

In this section, we present the experiment results for ℓ_2-regularized support vector machine (SVM) with smoothed hinge loss. Specifically, the objective function is $P(x) = \frac{1}{n}\sum_{i=1}^n \phi_i(a_i^\top x) + \frac{\lambda}{2}\|x\|_2^2$, where ϕ_i is defined in (3). For both the singly stochastic and doubly stochastic primal-dual frameworks, we compare our adaptive importance sampling (AIS) method with other sampling methods, i.e., stati-

onary Lipschitz-based importance sampling (LIS) and uniform sampling (US). All the algorithms are tested based on the real datasets a2a, w8a, gisette, and colon-cancer, where the the value of λ are set as $10^{-2}, 10^{-2}, 10^{-1}, 10^{0}$, respectively. The attributes of these datasets and values of λ chosen for each dataset are summarized in Table 2. The datasets basically cover three different types, i.e., $n \gg d$, $n \approx d$, and $n \ll d$.

Table 2: Parameters for Different Datasets

	colon-cancer	gisette	a2a	w8a
n	62	6000	2265	49746
d	2000	5000	123	300
λ	10^{0}	10^{-1}	10^{-2}	10^{-2}

4.1. Experiments on SPDC-AIS

We compare the performance of three algorithms, which are respectively SPDC-AIS, SPDC-LIS, and SPDC-US. For SPDC-AIS, we set $\kappa = 0.5$, $\underline{\delta} = 0.2$, $\overline{\delta} = 0.8$, and $\delta_t = \underline{\delta} + (\overline{\delta} - \underline{\delta})t/T$, where T is the maximum number of iterations we run. All the hyper-parameters τ, σ, θ are chosen by their theoretical values given in Theorem 1. For SPDC-LIS, we let the probability of sampling the i-th dual coordinate be $p_i^L = (1-\delta_t)\frac{1}{n} + \delta_t \frac{\|a_i\|_2}{\sum_{k=1}^n \|a_k\|_2}$, where $\|a_i\|_2$ is the Lipschitz constant of the component gradient $a_i \nabla \phi_i(a_i^\top x)$. As shown in Fig. 4, all three algorithms achieve linear convergence, while our proposed SPDC-AIS always exhibits notably sharper convergence rate than the other two algorithms. As for SPDC-LIS, it outperforms uniform sampling only in the case of colon-cancer dataset where $n \ll d$. For other three datasets, SPDC-LIS is just comparable with SPDC-US.

We also conduct experiments on two high dimensional datasets, i.e., RCV1 and Covtype. Besides, as suggested in (Zhang and Xiao, 2017), we choose three small regularization weights $\lambda = 10^{-4}, 10^{-6}, 10^{-8}$ to guarantee good accuracy. The results are reported in Table 3, which shows that our proposed SPDC-AIS always exhibits notably sharper convergence rate than other two algorithms.

4.2. Experiments on DSPDC-AIS & SPD1-VR-AIS

Similar to SPDC-based algorithms, we also test the performance of doubly stochastic algorithms. For DSPDC-AIS and SPD1-VR-AIS, we also set $\kappa = 0.5$, and $\delta_k = \underline{\delta} + (\overline{\delta} - \underline{\delta})k/K$, where K is the maximum number of epochs we run. We adopt best-tuned fixed stepsizes for the three algorithms. For DSPDC-LIS and SPD1-VR-LIS, we let the probability of sampling the i-th dual coordinate and j-th primal coordinate be respectively $p_i^L = (1-\delta_k)\frac{1}{n} + \delta_k \frac{\|a_i\|_2}{\sum_{k=1}^n \|a_k\|_2}$ and $q_j^L = (1-\delta_k)\frac{1}{n} + \delta_k \frac{\|\alpha_j\|_2}{\sum_{l=1}^d \|\alpha_l\|_2}$.

As shown in Fig. 5 and Fig. 6, our adaptive sampling-based algorithms converge much faster than the other two sampling methods for all datasets, while LIS-based algorithms has similar overall perfor-

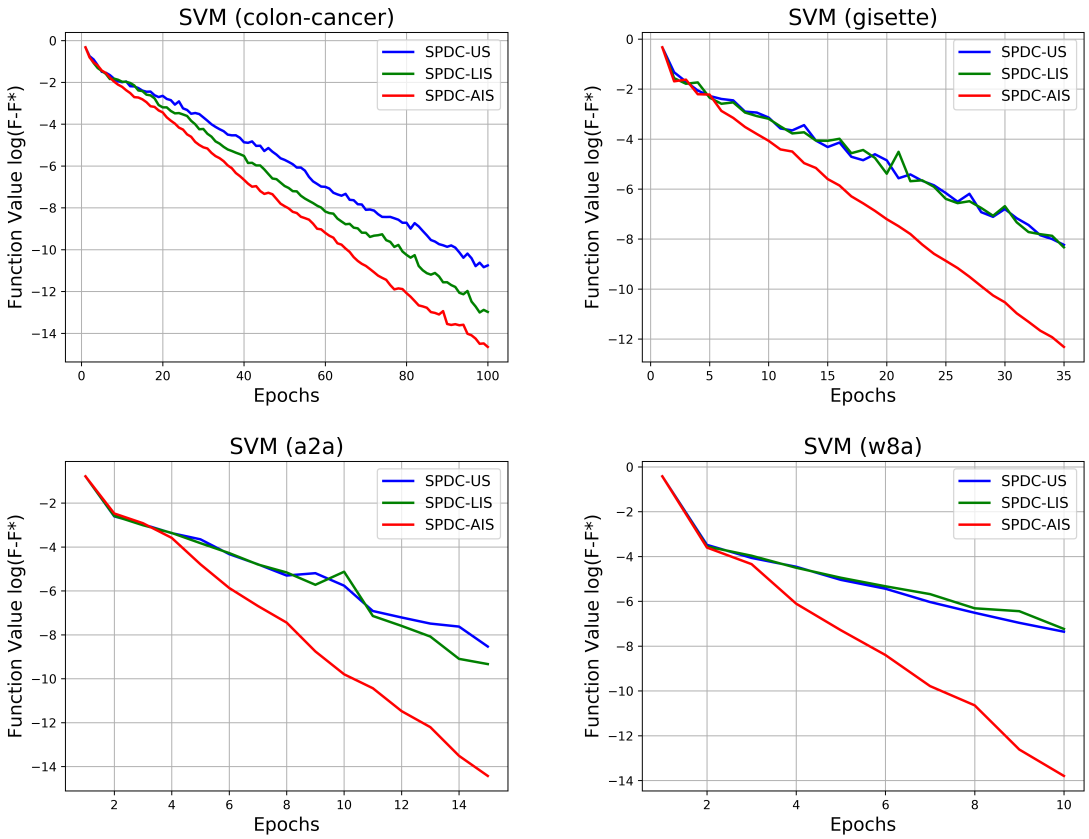


Fig. 4: Experiment Results for SPDC-based Algorithms

mance with US-based algorithms. These empirical results on real datasets demonstrate that our proposed adaptive importance sampling method noticeably accelerates the convergence of stochastic primal-dual algorithms.

4.3. Results of Execution Time

For fair comparison of the empirical results, we provide the execution time of the experiments in Sections 4.1 and 4.2. All our experiments are conducted based on an Intel i5 processor with 3.1GHz main frequency. Tables 2-4 present the specific running times (in seconds) per epoch. As expected, non-uniform sampling methods is somewhat more time-consuming than uniform sampling, since non-uniform sampling requires $O(\log n)$ time per iteration to generate a random number. Besides, adaptive sampling takes a little more time than Lipschitz-based sampling, since adaptive sampling requires extra $O(\log n)$ to update the binary tree. For the same dataset, the execution time does not differ much from each other.

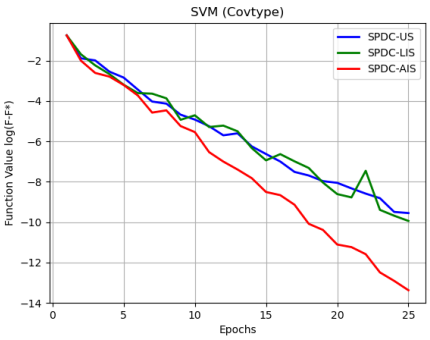
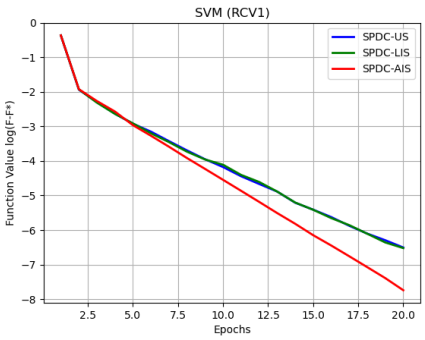
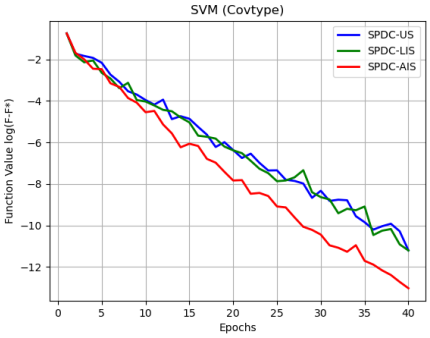
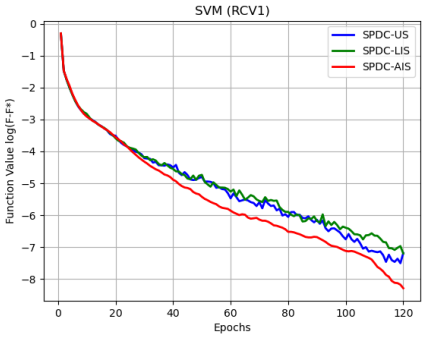
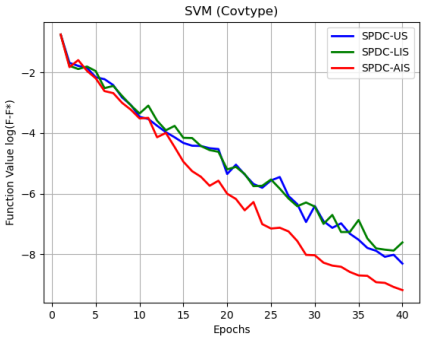
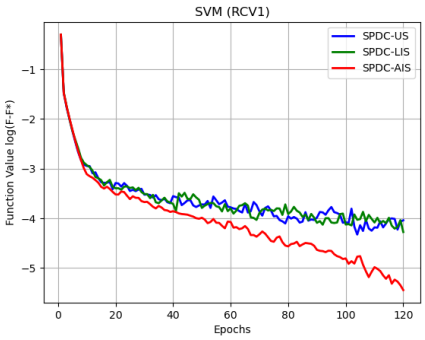
λ	Covtype	RCV1
10^{-4}		
10^{-6}		
10^{-8}		

Table 3: Experiment Results for SDPC-based Algorithms with Different Regularizer Weights

In particular, for datasets with relatively large dimension d, the extra execution time of our adaptive sampling is just marginal relative to the $O(d)$ time for updating the primal and dual variables. Together with the results in Fig. 4-6, we conclude that at the cost of slightly higher computational burden per epoch, our adaptive coordinate sampling methods significantly boost the stochastic primal-dual algorithms.

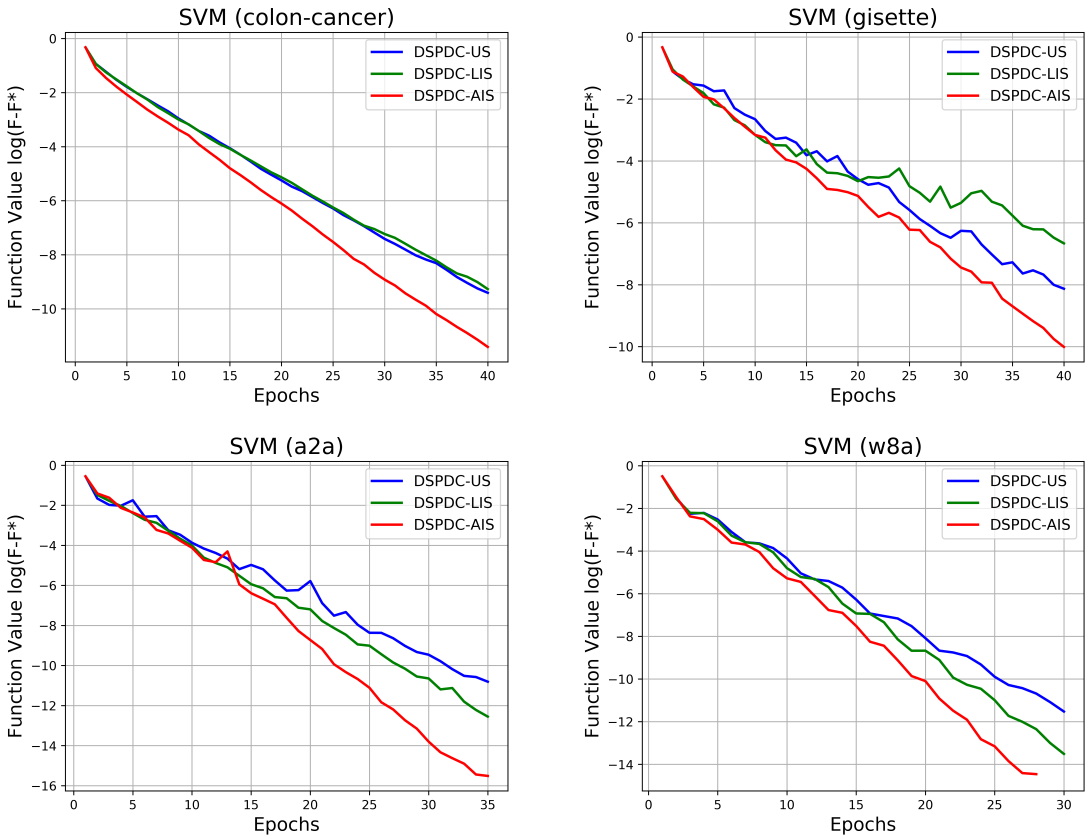


Fig. 5: Experiment Results for DSPDC-based Algorithms

Table 4: Execution Time of SPDC-based Algorithms

	colon-cancer	gisette	a2a	w8a
US	0.0042	3.01	0.0060	0.796
LIS	0.0043	3.09	0.0067	0.839
AIS	0.0049	3.34	0.008	0.895

5. Conclusion

In this paper, we have investigated an adaptive importance sampling method for stochastic primal-dual optimization algorithms. The proposed method samples the primal and dual coordinates by adapting to the local structure of the objective function. We have taken advantage of a specific binary tree structure to implement computationally efficient sampling. We have applied our sampling method to three com-

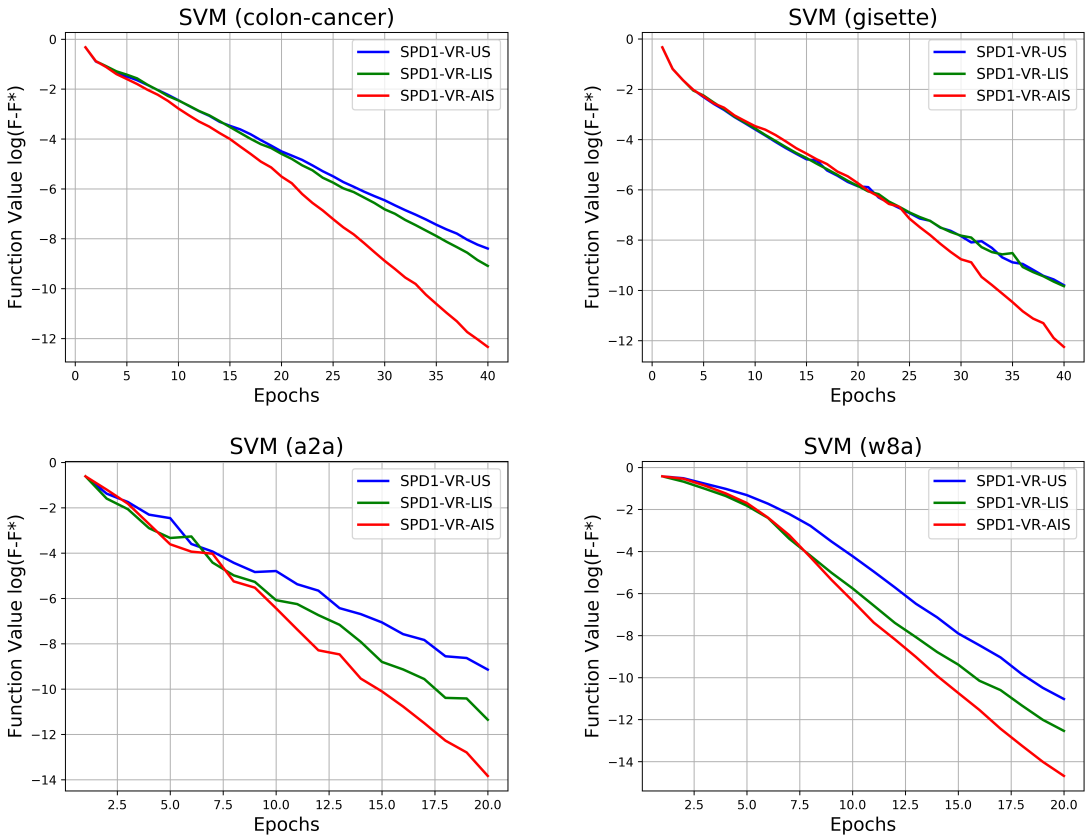


Fig. 6: Experiment Results for SPD1-based Algorithms

Table 5: Execution Time of DSPDC-based Algorithms

	colon-cancer	gisette	a2a	w8a
US	0.0150	2.05	0.0126	9.01
LIS	0.0185	2.21	0.0131	9.29
AIS	0.0195	2.30	0.0134	9.38

mon stochastic primal-dual algorithms, i.e., SPDC, DSPDC, and SPD1-VR. We have provided detailed theoretical analysis to justify the effectiveness of our methods. We have conducted experiments on real datasets verify that our adaptive coordinate sampling achieves significantly faster convergence than the common stationary sampling methods.

Table 6: Execution Time of SPD1-VR-based Algorithms

	colon-cancer	gisette	a2a	w8a
US	0.0080	5.936	0.0235	2.621
LIS	0.0120	6.033	0.0270	3.0425
AIS	0.0135	6.399	0.0295	3.1155

References

Allen-Zhu, Z., 2017. Katyusha: The first direct acceleration of stochastic gradient methods. *The Journal of Machine Learning Research* 18, 1, 8194–8244.

Allen-Zhu, Z., Qu, Z., Richtárik, P., Yuan, Y., 2016. Even faster accelerated coordinate descent using non-uniform sampling. In *International Conference on Machine Learning*, pp. 1110–1119.

Beck, A., Teboulle, M., 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences* 2, 1, 183–202.

Bottou, L., Curtis, F.E., Nocedal, J., 2018. Optimization methods for large-scale machine learning. *Siam Review* 60, 2, 223–311.

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

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

Csiba, D., Qu, Z., Richtárik, P., 2015. Stochastic dual coordinate ascent with adaptive probabilities. In *International Conference on Machine Learning*, pp. 674–683.

Csiba, D., Richtárik, P., 2018. Importance sampling for minibatches. *The Journal of Machine Learning Research* 19, 1, 962–982.

Defazio, A., Bach, F., Lacoste-Julien, S., 2014. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in neural information processing systems*, pp. 1646–1654.

Devroye, L., 1986. *Non-Uniform Random Variate Generation*. Springer.

Esser, E., Zhang, X., Chan, T.F., 2010. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM Journal on Imaging Sciences* 3, 4, 1015–1046.

Fang, C., Li, C.J., Lin, Z., Zhang, T., 2018. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pp. 689–699.

Hiriart-Urruty, J.B., Lemaréchal, C., 2012. *Fundamentals of convex analysis*. Springer Science & Business Media.

Horváth, S., Richtarik, P., 2019. Nonconvex variance reduced optimization with arbitrary sampling. In *International Conference on Machine Learning*, pp. 2781–2789.

Johnson, R., Zhang, T., 2013. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pp. 315–323.

Lin, Q., Lu, Z., Xiao, L., 2015. An accelerated randomized proximal coordinate gradient method and its application to regularized empirical risk minimization. *SIAM Journal on Optimization* 25, 4, 2244–2273.

Lu, Z., Xiao, L., 2015. On the complexity analysis of randomized block-coordinate descent methods. *Mathematical Programming* 152, 1-2, 615–642.

Namkoong, H., Sinha, A., Yadlowsky, S., Duchi, J.C., 2017. Adaptive sampling probabilities for non-smooth optimization. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, pp. 2574–2583.

Needell, D., Ward, R., Srebro, N., 2014. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. In *Advances in neural information processing systems*, pp. 1017–1025.

Nesterov, Y., 1998. Introductory lectures on convex programming volume i: Basic course. *Lecture notes* 3, 4, 5.

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

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

Nesterov, Y., Stich, S.U., 2017. Efficiency of the accelerated coordinate descent method on structured optimization problems. *SIAM Journal on Optimization* 27, 1, 110–123.

Papa, G., Bianchi, P., Clémençon, S., 2015. Adaptive sampling for incremental optimization using stochastic gradient descent. In *International Conference on Algorithmic Learning Theory*, Springer, pp. 317–331.

Perekrestenko, D., Cevher, V., Jaggi, M., 2017. Faster coordinate descent via adaptive importance sampling. In *Artificial Intelligence and Statistics*, pp. 869–877.

Qian, X., Richtarik, P., Gower, R., Sailanbayev, A., Loizou, N., Shulgin, E., 2019. SGD with arbitrary sampling: General analysis and improved rates. In *International Conference on Machine Learning*, pp. 5200–5209.

Qu, Z., Richtárik, P., 2016. Coordinate descent with arbitrary sampling i: Algorithms and complexity. *Optimization Methods and Software* 31, 5, 829–857.

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

Richtárik, P., Takáč, M., 2016. On optimal probabilities in stochastic coordinate descent methods. *Optimization Letters* 10, 6, 1233–1243.

Roux, N.L., Schmidt, M., Bach, F.R., 2012. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in neural information processing systems*, pp. 2663–2671.

Salehi, F., Thiran, P., Celis, E., 2018. Coordinate descent with bandit sampling. In *Advances in Neural Information Processing Systems*, pp. 9247–9257.

Shalev-Shwartz, S., 2016. Sdca without duality, regularization, and individual convexity. In *International Conference on Machine Learning*, PMLR, pp. 747–754.

Shalev-Shwartz, S., Tewari, A., 2011. Stochastic methods for l1-regularized loss minimization. *Journal of Machine Learning Research* 12, Jun, 1865–1892.

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

Stich, S.U., Raj, A., Jaggi, M., 2017. Safe adaptive importance sampling. In *Advances in Neural Information Processing Systems*, pp. 4381–4391.

Tan, C., Zhang, T., Ma, S., Liu, J., 2018. Stochastic primal-dual method for empirical risk minimization with o (1) per-iteration complexity. In *Advances in Neural Information Processing Systems*, pp. 8366–8375.

Tseng, P., 1998. An incremental gradient (-projection) method with momentum term and adaptive stepsize rule. *SIAM Journal on Optimization* 8, 2, 506–531.

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

Yu, A.W., Lin, Q., Yang, T., 2015. Doubly stochastic primal-dual coordinate method for bilinear saddle-point problem. *arXiv preprint arXiv:1508.03390*

Zhang, A., Gu, Q., 2016. Accelerated stochastic block coordinate descent with optimal sampling. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 2035–2044.

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

Zhao, P., Zhang, T., 2015. Stochastic optimization with importance sampling for regularized loss minimization. In *international conference on machine learning*, pp. 1–9.

Zhou, D., Xu, P., Gu, Q., 2018. Stochastic nested variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems* 31, 3921–3932.

Zhu, Z., Storkey, A.J., 2015. Adaptive stochastic primal-dual coordinate descent for separable saddle point problems. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, pp. 645–658.