Projection-based Regularized Dual Averaging for Stochastic Optimization

Asahi Ushio, *Student Member, IEEE,* Masahiro Yukawa, *Member, IEEE*

***Abstract*—We propose a novel stochastic-optimization framework based on the regularized dual averaging (RDA) method. The proposed approach differs from the previous studies of RDA in three major aspects. First, the squared-distance loss function to a 'random' closed convex set is employed for stability. Second, a sparsity-promoting metric (used implicitly by a certain proportionate-type adaptive filtering algorithm) and a quadratically-weighted ℓ_1 regularizer are used simultaneously. Third, the step size and regularization parameters are both constant due to the smoothness of the loss function. Those three differences yield an excellent sparsity-seeking property, high estimation accuracy, and insensitivity to the choice of the regularization parameter. Numerical examples show the remarkable advantages of the proposed method over the existing methods (including AdaGrad and the adaptive proximal forward-backward splitting method) in applications to regression and classification with real/synthetic data.**

***Index Terms*—online learning, regularized stochastic optimization, orthogonal projection, proximity operator**

I. Introduction

Sparse systems are encountered in many applications such as echo cancellation, channel estimation, text classification, etc. Here, "sparse" means that the system (Euclidean vector) contains many (nearly) zero components. Online estimation/learning of sparse systems can be formulated as a regularized stochastic optimization problem, where the goal is to minimize the expectation of stochastic loss function depending on random measurements (samples) penalized by a regularizer, such as the (weighted) ℓ_1 norm when the sparsity is preferred. We solely consider online scenarios in which the measurements arrive sequentially, although regularized stochastic optimization can also be considered in batch scenarios in general. *Stochastic dual averaging* [2, 3] is another popular stochastic-optimization method than the classical stochastic gradient descent (SGD) method, using subgradients of loss function. Meanwhile, the metric projection has been proven to be a powerful tool for adaptive filtering [4–6] as well as image processing, optics, among many others [7]. Nevertheless, its power for the stochastic dual averaging method still remains uninvestigated. The central question addressed in the present study is whether the metric projection also brings any benefits for the stochastic dual averaging method. To derive a projection-based dual-averaging method, we employ a squared-distance loss-function which is smooth. Smoothness of loss function is certainly advantageous from the optimization point of view, and it allows to use a constant step size. The use of constant step size actually makes essential differences (elaborated later on) from the original dual averaging framework, and it yields high sparsity and high estimation accuracy simultaneously.

Stochastic dual averaging has its origin in the work of Nesterov in deterministic settings [2], and it has been extended to stochastic settings by Xiao [3]. The stochastic version is called the regularized dual averaging (RDA) method, covering stochastic optimization problems involving a regularization term. The follow-the-regularized-leader is a similar approach to RDA, studied in the context of online convex optimization [8]. Meanwhile, Duchi and Singer have proposed the so-called FOBOS algorithm [9], which is an online extension of the celebrated proximal forward-backward splitting (also known as proximal gradient) method. It is an efficient solver for regularized stochastic optimization problems, including SGD as its special case. RDA typically produces a sparser solution than FOBOS when a sparsity-enhancing regularizer such as the ℓ_1 norm is involved, because RDA can use a more aggressive truncation threshold.

The history of using 'sparseness' in online algorithms traces back at least to the work of Makino and Kaneda in 1992 where the exponentially-decaying structure of acoustic echo path has been incorporated to an adaptive filtering algorithm [10, 11]. Duttweiller has proposed a related algorithm called the proportionate least mean square algorithm for acoustic echo cancellation problems [12–14]. Yukawa, Slavakis, and Yamada have shown in 2007 that those algorithms can be interpreted as changing the geometry of the Euclidean space by means of sparsity-promoting metrics (which vary in time) for improving the convergence behaviours [15]. The convergence properties of those algorithms have been studied in [16] under the framework of the adaptive projected subgradient method (APSM) [4, 5] with a variable-metric extension. Here, APSM asymptotically minimizes a sequence of nonnegative convex functions, and it gives a unified guiding principle of a wide range of adaptive algorithms including the normalized least mean square (NLMS) algorithm [17, 18], the affine projection algorithm (APA) [19, 20], the adaptive parallel subgradient projection algorithm [4], the multi-domain adaptive filtering algorithm [21], as well as their constrained counterparts [22]. The variable-metric APSM [16] further includes the transform-

This work was partially supported by JSPS Grants-in-Aid (18H01446, 15H02757). A preliminary version of this paper was presented at ICASSP 2017 [1].

This work was done while A. Ushio was with the Department of Electronics and Electrical Engineering, Keio University, Yokohama 223-8522, Japan. He is currently with Cogent Labs, Japan.

M. Yukawa is with the Department of Electronics and Electrical Engineering, Keio University, Yokohama 223-8522, Japan, and also with the RIKEN Center for Advanced Intelligence Project, Tokyo 103-0027, Japan (e-mail: yukawa@elec.keio.ac.jp).

domain adaptive filter [23–25], the LMS-Newton adaptive filter [26–28], the quasi-Newton adaptive filter [28–30], and the Krylov-proportionate adaptive filter [31–33].

Independently from those previous studies in signal processing community, a number of variable-metric stochastic-optimization algorithms have recently been developed, divided into three categories. The first category is the quasi-Newton type, including the finite-difference-method-based algorithms (e.g., SGD quasi-Newton [34], AdaDelta [35], and variance-based SGD [36, 37]), the extended Gauss-Newton-based algorithms [38–40], and the stochastic limited BFGS (Broyden Fletcher Goldfarb Shanno) algorithms [41–43]. These algorithms approximate the Hessian matrix efficiently. The second category is the natural-gradient type [44–46], approximating the Fisher information matrix. The third category exploits some metric based on the root-mean-square (RMS) of the history of the (sub)gradients, including AdaGrad [47], RMSprop [48], and Adam [49]. *The composite objective mirror descent (COMID) method* [50] is a generalization of FOBOS (in a wide sense), replacing the squared Euclidean distance term by a Bregman divergence to allow a use of non-Euclidean geometry. It reduces, for instance, to the exponentiated gradient method [51] when the Kullback-Leibler divergence is adopted.

Metric projection has played a key role in the success of many adaptive filtering algorithms [4, 5, 21], including all the algorithms that are raised above as particular examples of (variable-metric) APSM. It has also been used in the study of the adaptive proximal forward-backward splitting (APFBS) method proposed by Murakami, Yamagishi, Yukawa, and Yamada for optimizing a sequence of regularized objective functions [52, 53]. (A short summary of the projection-based method is given in Section II-C; see [6] for its comprehensive tutorial). NLMS operates iterative projections onto the zero-instantaneous-error hyperplanes. It often performs better and is more stable than the classical least mean square (LMS) algorithm [54], which is a SGD method for the mean squared error (MSE) function. It is natural to ask whether those projection and sparsity-promoting metrics studied extensively in signal processing community are useful in the RDA framework.

In this paper, we propose *the projection-based regularized dual averaging (PDA) method*, targeting 'sparsity-regularized' stochastic optimization problems primarily. The PDA method simultaneously exploits both a sparsity-promoting metric (changing the geometry) and a sparsity-promoting regularizer (shrinking the coefficients). This yields a remarkable sparsity-seeking-property, leading to high accuracy of classification/regression as well as low evaluation costs for validation data. The regularizer used here is referred to as the 'quadratically-weighted' ℓ_1 norm, and it is tailored to its simultaneous use with the sparsity-promoting metric. It is devised based on the interesting (and perhaps surprising) observation that the use of the typical weighted/unweighted ℓ_1 regularizer causes undesirable biases by shrinking the large coefficients to a larger extent than the smaller ones. The simultaneous use actually yields a certain synergy effect. In fact, the sole use of the sparsity-promoting metric typically causes slow convergence in the late learning-phase in return for fast initial convergence, since it increases the learning speed of the large coefficients but decreases that of the small ones. The sparsity-promoting regularizer attracts those small coefficients to zero, and this alleviates the slow-convergence issue as well as reducing the estimation variances efficiently (see Section III-C). Our loss function is the squared metric-distance to the random closed convex set, where the randomness comes from data/measurements. For instance, the zero-instantaneous-error hyperplane could be used for regression, and the instantaneous-discrimination-with-sufficient-margin halfspace for classification. As the loss function is smooth, PDA employs fixed step-size and regularization parameters, and this leads to remarkable insensitivity to the choice of the regularization parameter as shown by simulation. This makes PDA significantly different from the original RDA framework, in which the step size needs to diminish for ensuring convergence for possibly-nonsmooth stochastic optimization problems.

The major differences of PDA from the original RDA framework are the use of (cf. Section III-C):

1) the metric projection,
2) the step-size and regularization parameters both fixed in time, and
3) the sparsity-promoting metric together with the sparsity-promoting regularizer (the quadratically-weighted ℓ_1 norm to be specific).

These differences lead to three practical advantages in stochastic optimization involving sparse structures. First, the use of the squared-distance function stabilizes the algorithm, as it avoids such a situation that the gradient vector becomes undesirably large when some impulsive input arrives. Second, the simultaneous use of the metric and regularizer guides the update direction towards the true (sparse) solution and yields better bias-variance tradeoffs. Third, the use of a fixed regularization parameter prevents our estimates from becoming undesirably sparse. The efficacy of PDA is shown by simulations with the MNIST hand-written-digit and RCV text datasets for classification and with some acoustic signals as well as synthetic data for linear/nonlinear regression.

The rest of the paper is organized as follows. Section II introduces notation, the formulation of the regularized stochastic optimization problem, the RDA framework, and the projection-based methods. Section III presents the proposed method, its computational complexity for regression and classification, and the relations to prior works. Section IV presents simulation results, followed by conclusion in Section V.

II. Preliminaries

Throughout, the sets of real numbers, nonnegative integers, and positive integers are denoted by $\mathbb{R}$, $\mathbb{N}$, and $\mathbb{N}^*$, respectively. The standard inner product between $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{z} \in \mathbb{R}^n$ is defined as $\langle \boldsymbol{w}, \boldsymbol{z} \rangle := \boldsymbol{w}^\mathsf{T}\boldsymbol{z}$, where the superscript $(\cdot)^\mathsf{T}$ stands for transpose. The Euclidean norm of $\boldsymbol{w} \in \mathbb{R}^n$ is denoted by $\|\boldsymbol{w}\| := \sqrt{\langle \boldsymbol{w}, \boldsymbol{w} \rangle}$. Given a positive definite matrix $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$, the $\boldsymbol{Q}$ inner product between $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{z} \in \mathbb{R}^n$ is defined as $\langle \boldsymbol{w}, \boldsymbol{z} \rangle_{\boldsymbol{Q}} := \boldsymbol{w}^\mathsf{T}\boldsymbol{Q}\boldsymbol{z}$. The $\boldsymbol{Q}$-norm is defined as $\|\boldsymbol{w}\|_{\boldsymbol{Q}} := \sqrt{\langle \boldsymbol{w}, \boldsymbol{w} \rangle_{\boldsymbol{Q}}}$. For a differentiable convex function $f : \mathbb{R}^n \to \mathbb{R}$, a vector $\nabla_{\boldsymbol{Q}} f(\boldsymbol{w}) \in \mathbb{R}^n$ is called the $\boldsymbol{Q}$-gradient of f at $\boldsymbol{w}$ if $\left\langle \boldsymbol{z} - \boldsymbol{w}, \nabla_{\boldsymbol{Q}} f(\boldsymbol{w}) \right\rangle_{\boldsymbol{Q}} +$

$f(\boldsymbol{w}) \le f(\boldsymbol{z})$ for any $\boldsymbol{z} \in \mathbb{R}^n$. The identity matrix is denoted by $\boldsymbol{I}$. The ordinary gradient operator $\nabla_{\boldsymbol{I}}$ is denoted simply by ∇. An operator $T : \mathbb{R}^n \to \mathbb{R}^n$ is said to be Lipschitz continuous if there exists a constant $\mu > 0$ such that $\|T(\boldsymbol{x}) - T(\boldsymbol{y})\|_{\boldsymbol{Q}} \le \mu \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{Q}}$ for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$.

A. Problem Formulation

The regularized stochastic optimization problems considered in this paper are stated as follows:

$$\min_{\boldsymbol{w}\in\mathbb{R}^n} \mathbb{E}_z \left[\varphi(\boldsymbol{w}, z)\right] + \psi(\boldsymbol{w}), \tag{1}$$

where $\mathbb{E}_z$ stands for expectation with respect to the input-output pair $z := (\boldsymbol{x}, y) \in \mathbb{R}^n \times \mathbb{R}$ drawn from an unknown underlying distribution, and the loss function $\varphi(\boldsymbol{w}, z)$ and the regularizer $\psi(\boldsymbol{w})$ are both assumed convex. A typical role of the regularizer $\psi(\boldsymbol{w})$ is promoting the sparsity of solution, and it is our primal interest in the present study as well. As the expectation is unavailable in practice, we consider the following empirical loss at each time instant $t \in \mathbb{N}$ penalized by the *time-dependent* regularizer $\psi_t(\boldsymbol{w})$:

$$\min_{\boldsymbol{w}\in\mathbb{R}^n} \frac{1}{t} \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w})\right] + \psi_t(\boldsymbol{w}), \tag{2}$$

where $\varphi_\tau(\boldsymbol{w}) := \varphi(\boldsymbol{w}, z_\tau)$ with the observation $z_\tau := (\boldsymbol{x}_\tau, y_\tau) \in \mathbb{R}^n \times \mathbb{R}$ of z at time $\tau = 1, 2, \cdots, t$. Here, the time-dependency of $\psi_t(\boldsymbol{w})$ is for suppressing an undesirable increase of estimation biases, as will be clarified in Section III. The weight vector (the coefficient vector) at an arbitrary time τ is denoted by $\boldsymbol{w}_\tau := [w_{\tau,1}, w_{\tau,2}, \cdots, w_{\tau,n}]^{\mathsf{T}} \in \mathbb{R}^n$.

B. Regularized Dual Averaging

The stochastic dual averaging method seeks to solve (1) efficiently for $\psi = 0$, and it is based on the following formulation[1] [2]:

$$\begin{aligned} \min_{\boldsymbol{w}\in\mathbb{R}^n} \; & l_t(\boldsymbol{w}) := \frac{1}{t} \sum_{\tau=1}^{t} \left[\varphi_\tau(\boldsymbol{w}_\tau) + \langle \nabla\varphi_\tau(\boldsymbol{w}_\tau), \boldsymbol{w} - \boldsymbol{w}_\tau \rangle\right] \\ & \text{subject to} \;\; h(\boldsymbol{w}) \le D, \end{aligned} \tag{3}$$

where $h(\boldsymbol{w})$ is a strongly-convex continuous function (a prox-function), and $D > 0$. The role of $h(\boldsymbol{w})$ is changing the geometry of space, and a typical, and also simplest, choice is $h(\boldsymbol{w}) := \|\boldsymbol{w}\|^2/2$. The function $l_t(\boldsymbol{w})$ is called *the lower linear model*, and it is actually an average of the affine minorants of $\varphi_\tau(\boldsymbol{w})$. The dual averaging update is given by

$$\begin{aligned} \boldsymbol{w}_t :&= \arg\min_{\boldsymbol{w}\in\mathbb{R}^n} \left(l_t(\boldsymbol{w}) + \frac{\beta_t}{t} h(\boldsymbol{w}) \right) \\ &= \arg\min_{\boldsymbol{w}\in\mathbb{R}^n} \left(\left\langle \frac{\boldsymbol{s}_t^{\boldsymbol{I}}}{t}, \boldsymbol{w} \right\rangle + \frac{\beta_t}{t} h(\boldsymbol{w}) \right), \end{aligned} \tag{4}$$

where $(\beta_\tau)_{\tau=1}^t \in [0, \infty)$ is a nonnegative and non-decreasing sequence, and

$$\boldsymbol{s}_t^{\boldsymbol{I}} := \sum_{\tau=1}^{t} \nabla\varphi_\tau(\boldsymbol{w}_\tau). \tag{5}$$

[1] In the original paper [2], a time-invariant function was used in place of φ_τ since deterministic optimization was considered there.

Here, $(\beta_\tau)_{\tau=1}^t$ wants to satisfy $\lim_{t\to\infty} \beta_t/t = 0$ so that the impact of $h(\boldsymbol{w})$ diminishes as $t \to \infty$.

In the RDA framework, the regularizer is considered to be time-invariant, i.e., $\psi_t := \psi$ for all $t = 1, 2, \cdots$, and an update equation is given as [3]

$$\boldsymbol{w}_t := \arg\min_{\boldsymbol{w}\in\mathbb{R}^n} \left(\left\langle \frac{\boldsymbol{s}_t^{\boldsymbol{I}}}{t}, \boldsymbol{w} \right\rangle + \frac{\beta_t}{t} h(\boldsymbol{w}) + \psi(\boldsymbol{w}) \right). \tag{6}$$

C. Projection-based Method

We consider the unregularized case (i.e., the case of $\psi_t := 0$). For a time-variant positive definite matrix $\boldsymbol{Q}_t$, the $\boldsymbol{Q}_t$-metric distance from an arbitrary point $\boldsymbol{w} \in \mathbb{R}^n$ to a closed convex set $C_t \subset \mathbb{R}^n$ is defined as $d_{\boldsymbol{Q}_t}(\boldsymbol{w}, C_t) := \min_{\boldsymbol{z}\in C_t} \|\boldsymbol{w} - \boldsymbol{z}\|_{\boldsymbol{Q}_t}$. A projection-based method typically employs the squared-distance loss function[2] [4–6, 15, 16, 21]:

$$\varphi_t(\boldsymbol{w}) := \frac{1}{2} d_{\boldsymbol{Q}_t}^2(\boldsymbol{w}, C_t). \tag{7}$$

The squared-distance loss is common to both regression and classification. The design of the set C_t is task-dependent (see Section III-B). The $\boldsymbol{Q}_t$-gradient of φ_t at the previous point $\boldsymbol{w}_{t-1} \in \mathbb{R}^n$ is given by

$$\boldsymbol{g}_t := \nabla_{\boldsymbol{Q}_t}\varphi_t(\boldsymbol{w}_{t-1}) = \boldsymbol{w}_{t-1} - P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}), \tag{8}$$

where $P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}) := \arg\min_{\boldsymbol{z}\in C_t} \|\boldsymbol{w} - \boldsymbol{z}\|_{\boldsymbol{Q}_t}$ is the $\boldsymbol{Q}_t$-projection of $\boldsymbol{w}$ onto C_t. The SGD update is given by

$$\boldsymbol{w}_t := \boldsymbol{w}_{t-1} - \eta \boldsymbol{g}_t, \quad \eta > 0. \tag{9}$$

The projection-based method enjoys two major advantages. The first advantage is that the step size tuning is simple because of the nonexpansivity (i.e., the Lipschitz continuity with constant 1) of the gradient operator $\nabla_{\boldsymbol{Q}_t}\varphi_t : \mathbb{R}^n \to \mathbb{R}^n$ in the $\boldsymbol{Q}_t$-norm sense. Indeed, the allowed step-size range is simply given by $\eta \in [0, 2]$, whereas the upper bound in the case of the ordinary least-square (squared errors) loss

$$\varphi_t^{\mathrm{LS}}(\boldsymbol{w}) := \left(y_t - \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t\right)^2/2 \tag{10}$$

depends on the eigenvalues of the input autocorrelation matrix.

The second advantage is numerical stability (robustness against impulsive inputs). To illustrate this, let us consider the hyperplane

$$C_t := \left\{ \boldsymbol{w} \in \mathbb{R}^n \mid \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t = y_t \right\}, \tag{11}$$

which is widely used for online regression (or adaptive filtering). Here, $\boldsymbol{x}_t \in \mathbb{R}^n$ is the input vector at time instant t, $y_t := \boldsymbol{w}_*^{\mathsf{T}}\boldsymbol{x}_t + \nu_t \in \mathbb{R}$ is the output with the unknown vector $\boldsymbol{w}_* := [w_{*,1}, w_{*,2}, \cdots, w_{*,n}]^{\mathsf{T}} \in \mathbb{R}^n$ and the additive noise $\nu_t \in \mathbb{R}$. In this case, the squared-distance function in (7) reduces to the normalized least-square loss

$$\varphi_t^{\mathrm{NLS}}(\boldsymbol{w}) := \frac{\left(y_t - \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t\right)^2}{2\left\|\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t\right\|_{\boldsymbol{Q}_t}^2}, \;\; \boldsymbol{w} := [w_1, w_2, \cdots, w_n]^{\mathsf{T}} \in \mathbb{R}^n. \tag{12}$$

[2] Although it is irrelevant to the current work, *the parallel projection* was employed in the previous works [4–6, 15, 16, 21] to accelerate the convergence efficiently.

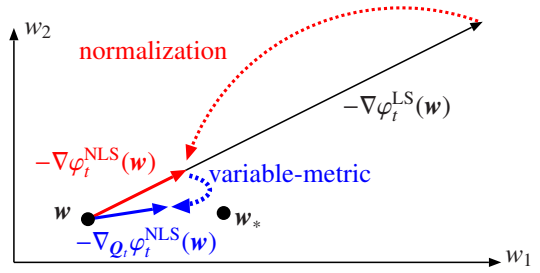


Fig. 1: Anti-gradients for an impulsive input vector.

TABLE I: PDA algorithm.

Requirements: $\lambda > 0$, $\eta \in [0, 2]$, $\alpha \in [0, 1]$, $\epsilon > 0$
($r \in \mathbb{N}^*$, $\delta > 0$ for regression ▷Section III-B)
Initialization: $\boldsymbol{s}_0 := \boldsymbol{0} \in \mathbb{R}^n$ and $\boldsymbol{w}_0 := \boldsymbol{0} \in \mathbb{R}^n$.
Iteration: For $t = 1, 2, \cdots$
1. $\boldsymbol{g}_t := \boldsymbol{w}_{t-1} - P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1})$ ▷ See (22) or (25).
2. $\boldsymbol{s}_t := \boldsymbol{s}_{t-1} + \boldsymbol{g}_t$
3. $\boldsymbol{w}_t := \mathrm{prox}_{\psi_t}^{\boldsymbol{Q}_t}(-\eta \boldsymbol{s}_t)$ ▷ See (16) with $\omega_{t,i} := q_{t,i}^2$.
4. Update $\boldsymbol{Q}_t$ ▷ See (14).

Figure 1 illustrates three anti-gradient vectors: $-\nabla\varphi_t^{\mathrm{LS}}(\boldsymbol{w})$, $-\nabla\varphi_t^{\mathrm{NLS}}(\boldsymbol{w})$, and $-\nabla_{\boldsymbol{Q}_t}\varphi_t^{\mathrm{NLS}}(\boldsymbol{w})$. The gradient $\nabla\varphi_t^{\mathrm{LS}}(\boldsymbol{w})$ is sensitive to impulsive inputs, and this causes instability. In contrast, $\nabla\varphi_t^{\mathrm{NLS}}(\boldsymbol{w})$, and $\nabla_{\boldsymbol{Q}_t}\varphi_t^{\mathrm{NLS}}(\boldsymbol{w})$ are robust to impulsive inputs due to the presence of the normalization factor. The metric $\boldsymbol{Q}_t$, in addition, guides the update direction towards the optimal point $\boldsymbol{w}_*$, leading to convergence acceleration. To see it, we inspect the $\boldsymbol{Q}_t$-projection for the specific C_t given in (11):

$$P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}) := \boldsymbol{w}_{t-1} - \frac{\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t - y_t}{\left\|\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t\right\|_{\boldsymbol{Q}_t}^2}\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t. \tag{13}$$

The update direction here is given by $\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t$ (or $-\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t$), while the ordinary SGD update direction is given by $\boldsymbol{x}_t$ (or $-\boldsymbol{x}_t$). This means that the update direction is changed by the *inverse* matrix $\boldsymbol{Q}_t^{-1}$ when the $\boldsymbol{Q}_t$-metric is adopted. Intuitively, when $\boldsymbol{w}_*$ is sparse and $|w_{*,1}| \gg |w_{*,2}|$, the step size in the w_1 direction needs to be larger than that in the w_2 direction, provided that $\boldsymbol{w}_0 := \boldsymbol{0}$. In this case, by allocating a larger step size to w_1 than w_2, or in other words by letting $\boldsymbol{Q}_t := \mathrm{diag}(q_1, q_2)$ with $0 < q_1 \ll q_2$ ($\Leftrightarrow q_1^{-1} \gg q_2^{-1} > 0$), a better direction of update can be obtained.

III. Projection-based Regularized Dual Averaging

The proposed PDA framework is presented in Section III-A. Under the use of a sparsity-promoting metric, an adequately-weighted ℓ_1-norm is devised that enhances sparsity without causing serious biases. The complexity of PDA for regression and classification is discussed in Section III-B. The advantages of PDA and its relation to RDA are discussed in Section III-C, and the relations to the other prior works (projection-based methods, AdaGrad [47], and APFBS [52, 53]) are discussed in Section III-D.

A. Proposed Method

The proposed method employs some sparsity-promoting metric. Recalling the arguments in Section II-C, one may want to use $\tilde{\boldsymbol{Q}}_t := (\mathrm{diag}(|w_{t-1,1}|, |w_{t-1,2}|, \cdots, |w_{t-1,n}|) + \epsilon\boldsymbol{I})^{-1}$, assuming that $\boldsymbol{w}_{t-1}$ well approximates $\boldsymbol{w}_*$, where the small constant $\epsilon > 0$ prevents devision by zero. The metric

$$\boldsymbol{Q}_t := \mathrm{diag}(q_{t,1}, q_{t,2}, \cdots, q_{t,n}) = \alpha\boldsymbol{I} + (1-\alpha)\frac{n\tilde{\boldsymbol{Q}}_t}{\mathrm{trace}\tilde{\boldsymbol{Q}}_t} \tag{14}$$

is based on the so-called *metric-combining* technique [55, 56], and it gives a better performance in practice, where $\alpha \in [0, 1]$.[3] See Section III-C for discussions about the metric.

Let us consider the use of a weighted ℓ_1 regularizer:

$$\psi_t^{\omega}(\boldsymbol{w}) := \lambda \sum_{i=1}^{n} \omega_{t,i}|w_i|, \tag{15}$$

where $\lambda > 0$ is the regularization parameter, and $\omega_{t,i} > 0$ is the weight assigned to w_i at time t. The $\boldsymbol{Q}_t$-proximity operator of $\psi_t^{\omega}(\boldsymbol{w})$ in (15) is then given by

$$\begin{aligned}\mathrm{prox}_{\psi_t^{\omega}}^{\boldsymbol{Q}_t}(\boldsymbol{w}) &:= \arg\min_{\boldsymbol{z}\in\mathbb{R}^n}\left(\psi_t^{\omega}(\boldsymbol{w}) + \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{z}\|_{\boldsymbol{Q}_t}^2\right)\\ &= \sum_{i=1}^{n} \boldsymbol{e}_i \mathrm{sgn}(w_i)\left[|w_i| - \omega_{t,i}q_{t,i}^{-1}\lambda\right]_+,\end{aligned} \tag{16}$$

where $\{\boldsymbol{e}_i\}_{i=1}^n$ is the standard basis of $\mathbb{R}^n$, $\mathrm{sgn}(\cdot)$ is the signum function, and $[\cdot]_+ := \max\{\cdot, 0\}$ is the hinge function.

Let us see how the proximity operator behaves when $\omega_{t,i} := 1$ for all $i = 1, 2, \cdots, n$ for a given time instant t; this is the case of the unweighted ℓ_1 norm $\|\boldsymbol{w}\|_1 := \sum_{i=1}^n |w_i|$. In this case, a simple inspection of (16) implies that the proximity operator shrinks the larger components to a larger extent than the smaller ones, because $q_{t,i}^{-1}$ is an increasing function of $|w_{t-1,i}|$. This actually causes undesirable biases. To mitigate the biases, we propose to use the following 'quadratically-weighted' ℓ_1 regularizer ($\omega_{t,i} := q_{t,i}^2$):

$$\psi_t(\boldsymbol{w}) = \lambda\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1} := \lambda\sum_{i=1}^{n} q_{t,i}^2|w_i|. \tag{17}$$

In this case, we have $\omega_{t,i}q_{t,i}^{-1} = q_{t,i}$ (which is a decreasing function of $|w_{t-1,i}|$).[4] As a result, the large components are kept less distorted, while the small components are attracted to zero efficiently. In the extreme case of $\alpha := 1$, the metric reduces to the Euclidean one $\boldsymbol{Q}_t = \boldsymbol{I}$ and the regularizer ψ_t to the ordinary ℓ_1 norm (up to a constant).

To understand the relation between the metric and the proximity operator intuitively, we illustrate in Figure 2 the unit balls for the three norms: $\|\boldsymbol{w}\|_1$, $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t,1} := \sum_{i=1}^n q_{t,i}|w_i|$, and $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$. Figure 2a shows that $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t,1}$ (or $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$) gives a 'fat'-shaped unit ball. This means that a large $|w_1|$ and some

[3] The metric-combining idea appearing in (14) is different from that of the improved proportionate NLMS algorithm [57] which uses the matrix $(\boldsymbol{Q}_t^{\mathrm{IP}})^{-1} := \alpha_{\mathrm{IP}}\boldsymbol{I} + (1-\alpha_{\mathrm{IP}})n\tilde{\boldsymbol{Q}}_t^{-1}/\mathrm{trace}\tilde{\boldsymbol{Q}}_t^{-1}$, $\alpha_{\mathrm{IP}} \in (0, 1)$, essentially. It has been shown in [56] that $\boldsymbol{Q}_t$ in (14) is more controllable than $\boldsymbol{Q}_t^{\mathrm{IP}}$ in the sense that, as the parameter α (or α_{IP}) changes gradually, the performance changes gradually as well.

[4] In our previous work in [58], we studied $\omega_{t,i}^{\mathrm{prev}} := \tilde{\omega}_{t,i}/(\sum_{j=1}^n \tilde{\omega}_{t,j})$ in the APFBS framework under the standard Euclidean metric (which is equivalent to $\boldsymbol{Q}_t := \frac{1}{n}\boldsymbol{I}$), where $\tilde{\omega}_{t,i} := (|w_{t-1,i}|^{1-p} + \epsilon)^{-1}$ with $\epsilon > 0$ and $p \in (0, 1)$, and $p = 0$ gave the best performance (see also [59–61]). In the present case, the particular choice of $\alpha := 0$ makes $\omega_{t,i}q_{t,i}^{-1} = q_{t,i} = \omega_{t,i}^{\mathrm{prev}}$ for $p = 0$. This is the reason behind the adoption of the quadratic weights in (17).

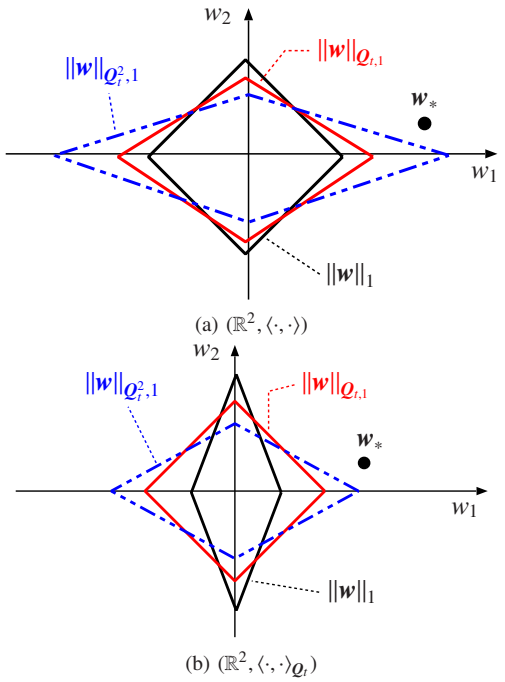


Fig. 2: Unit weighted-ℓ_1-balls under different geometries.

small $|w_2|$ takes the same penalty. Hence, compared to $\|\boldsymbol{w}\|_1$, the use of $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t,1}$ (or $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$) permits large components to stay large. Let us turn our attention to Figure 2b. One can see that, in the $\boldsymbol{Q}_t$ geometry, the unit ball associated with $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t,1}$ is a square and no longer 'fat'-shaped. On the other hand, the quadratically-weighted ℓ_1 norm $\|\boldsymbol{w}\|_{\boldsymbol{Q}_t^2,1}$ still possesses a 'fat' shape, thereby avoiding the undesirable biases.

For the initial vector $\boldsymbol{w}_0 := \boldsymbol{0}$, the proposed method is given as follows:

$$\begin{aligned}\boldsymbol{w}_t :&= \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\left(\langle \eta\boldsymbol{s}_t, \boldsymbol{w}\rangle_{\boldsymbol{Q}_t} + \frac{1}{2}\|\boldsymbol{w}\|_{\boldsymbol{Q}_t}^2 + \psi_t(\boldsymbol{w})\right)\\ &= \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\left(\psi_t(\boldsymbol{w}) + \frac{1}{2}\|\boldsymbol{w}+\eta\boldsymbol{s}_t\|_{\boldsymbol{Q}_t}^2\right)\\ &= \mathrm{prox}_{\psi_t}^{\boldsymbol{Q}_t}(-\eta\boldsymbol{s}_t), \quad t\in\mathbb{N}, \end{aligned} \tag{18}$$

where $\eta \in [0, 2]$ and

$$\boldsymbol{s}_t := \sum_{\tau=1}^{t} \boldsymbol{g}_\tau \ (= \boldsymbol{s}_{t-1} + \boldsymbol{g}_t), \quad t\in\mathbb{N}. \tag{19}$$

Here, we let $\boldsymbol{s}_0 := \boldsymbol{0}$. The proposed PDA method is summarized in Table I. More discussions are left to Section III-C.

B. Computational Complexity

The computational complexity (the number of multiplications per iteration) of each operation of the PDA method is given as follows: nr for the error calculations, nr^2 for the input normalization, $r^2 + nr$ for the $\boldsymbol{w}_t$ updates, $2n$ for the proximity-operator calculations, and another $2n$ for the metric calculations. The whole per-iteration complexity is actually governed by that of $P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1})$, and it is $O(n)$ as long as the set C_t is sufficiently simple. The design of C_t (and hence the complexity of PDA) depends on tasks as mentioned already, and more detailed discussions are given below for the regression and classification cases.

TABLE II: Computational complexity.

Algorithms	number of multiplication
PDA, APFBS	$4n + 3nr + nr^2 + r^2$ ($O(n)$ if $r = 1$)
RDA, AdaGrad, and FOBOS	$O(n)$

1) Regression case: We define the linear variety

$$C_t := \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\ (\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w} - \boldsymbol{y}_t)^{\mathsf{T}}(\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w} - \boldsymbol{y}_t), \tag{20}$$

where $\boldsymbol{X}_t := [\boldsymbol{x}_t\ \boldsymbol{x}_{t-1}\cdots\boldsymbol{x}_{t-r+1}] \in \mathbb{R}^{n\times r}$ and $\boldsymbol{y}_t := [y_t, y_{t-1}, \cdots, y_{t-r+1}]^{\mathsf{T}} \in \mathbb{R}^r$ for some $r \in \mathbb{N}^*$. The squared-distance function reduces to

$$\varphi_t(\boldsymbol{w}) = \frac{1}{2}[\boldsymbol{X}_t^{\dagger}(\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w} - \boldsymbol{y}_t)]^{\mathsf{T}}\boldsymbol{Q}_t^{-1}\boldsymbol{X}_t^{\dagger}(\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w} - \boldsymbol{y}_t), \tag{21}$$

and the $\boldsymbol{Q}_t$-projection is given by

$$P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}) = \boldsymbol{w}_{t-1} - \boldsymbol{Q}_t^{-1}\boldsymbol{X}_t^{\dagger}(\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{w}_{t-1} - \boldsymbol{y}_t), \tag{22}$$

where $\boldsymbol{X}_t^{\dagger}$ is the Moore-Penrose pseudo-inverse. In practice, $\boldsymbol{X}_t^{\dagger}$ is replaced by $\boldsymbol{X}_t(\boldsymbol{X}_t^{\mathsf{T}}\boldsymbol{Q}_t^{-1}\boldsymbol{X}_t + \delta\boldsymbol{I})^{-1}$, where $\delta > 0$ is the regularization parameter for numerical stability. For $r = 1$, C_t reduces to the hyperplane defined in (11).

The overall complexity for the regression case is $(r^2 + 3r + 4)n + r^2$, which is the same as APFBS. In the particular case $\boldsymbol{Q}_t = \boldsymbol{I}$, the complexity of PDA is reduced to $(r^2+2r+1)n+r^2$. Since r is typically small ($r = 1$ or $r = 2$ in our simulations), the complexity is $O(n)$ basically.

2) Classification case: We define the halfspace

$$C_t := \left\{\boldsymbol{w}\in\mathbb{R}^n \mid y_t\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t \geq 1\right\}, \tag{23}$$

where $y_t \in \{-1, 1\}$ ($\boldsymbol{x}_t \neq \boldsymbol{0}$ is assumed here). The squared-distance function in (7) then reduces to

$$\varphi_t(\boldsymbol{w}) = \frac{\left(\left[y_t\boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}_t - 1\right]_+\right)^2}{2\left\|\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t\right\|_{\boldsymbol{Q}_t}^2}. \tag{24}$$

The $\boldsymbol{Q}_t$-projection onto C_t is given in this case by

$$P_{C_t}^{\boldsymbol{Q}_t}(\boldsymbol{w}_{t-1}) = \boldsymbol{w}_{t-1} - \frac{\left[1 - y_t\boldsymbol{w}_{t-1}^{\mathsf{T}}\boldsymbol{x}_t\right]_+}{y_t\left\|\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t\right\|_{\boldsymbol{Q}_t}^2}\boldsymbol{Q}_t^{-1}\boldsymbol{x}_t. \tag{25}$$

The overall complexity for the classification case is $8n + 1$ ($4n + 1$ in the particular case of $\boldsymbol{Q}_t = \boldsymbol{I}$). The other regularized stochastic optimization methods such as AdaGrad, RDA, and FOBOS also have $O(n)$ complexity.

C. Advantages of PDA: Relation to RDA

The RDA framework in (6) can be rewritten as

$$\boldsymbol{w}_t := \underset{\boldsymbol{w}\in\mathbb{R}^n}{\arg\min}\left\{\left\langle \boxed{\frac{1}{\beta_t}}\,\boldsymbol{s}_t, \boldsymbol{w}\right\rangle + h(\boldsymbol{w}) + \boxed{\frac{t}{\beta_t}}\,\psi(\boldsymbol{w})\right\}. \tag{26}$$

Here, $1/\beta_t$ can be regarded as the step size, and t/β_t governs the strength of the regularization. For comparison, we consider the ordinary ℓ_1 regularizer $\psi(\boldsymbol{w}) := \lambda\|\boldsymbol{w}\|_1 := \lambda\sum_{i=1}^n |w_i|$ for

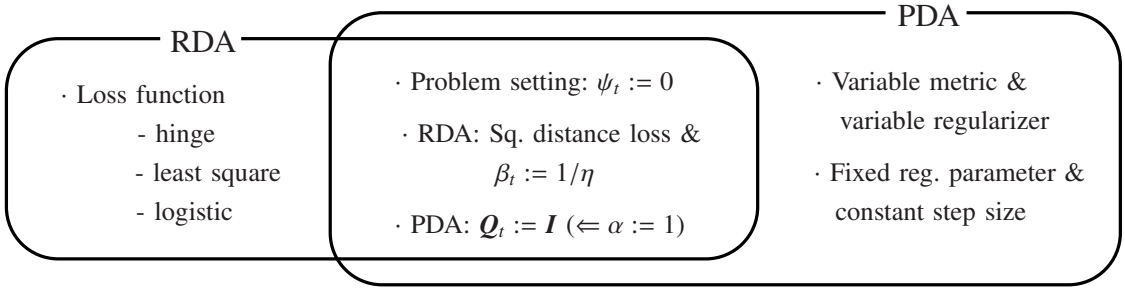


Fig. 3: A relation betwen RDA and PDA ($\psi_t \neq 0 \Rightarrow$ RDA $\neq$ PDA).

some fixed $\lambda > 0$. In the RDA framework, an increasing sequence $(\beta_\tau)_{\tau=1}^{\infty}$ needs to be used so that the step-size sequence $(1/\beta_\tau)_{\tau=1}^{\infty}$ diminishes, since possibly nonsmooth loss functions are considered [3]. The typical choice $\beta_t = \sqrt{t}/\eta$ for some $\eta > 0$ makes $t/\beta_t = \eta\sqrt{t} \sim O(\sqrt{t})$, which means that the strength of regularization increases at the rate of $O(\sqrt{t})$. In the present study, on the other hand, there is no need to use diminishing step size, since our loss function is smooth. The use of constant step size $1/\beta_t = \eta$, however, makes $t/\beta_t = \eta t \sim O(t)$, which increases linearly in t. As a result, the regularization becomes strong much faster than the case of diminishing step size, and thus the performance of RDA deteriorates severely unless the regularization parameter λ is tuned carefully. Indeed, the performance of RDA with constant step size is sensitive to the choice of λ, as shown in Section IV-A.

The key ideas and major advantages of the proposed PDA method are summarized as below.

a) The use of the squared-distance function given in (7) leads directly to the projection used in PDA. It enables to use constant step size and also brings robustness to impulsive inputs (stable learning), as mentioned already in Section II-C.
b) The quadratically-weighted ℓ_1 regularizer avoids the undesirable biases under the use of the sparsity-promoting metric, as elaborated in Section III-A. We emphasize here that PDA employs two sparsity-aware techniques (i.e., the metric and the regularizer) simultaneously. The metric accelerates the (initial) convergence but unchanges the optimal point indeed. Meanwhile, the regularizer seeks to decrease the MSE by reducing the estimation variance at the price of a slight increase of the estimation bias. The proposed method benefits from both. The sparsity-promoting metric actually encourages the learning of those coefficients with large magnitudes but discourages the learning of those with small magnitudes. This causes slow convergence at the final learning phase, and the parameter α introduced in (14) alleviates such a negative effect. In addition, the regularizer attracts those minor coefficients to zero, and thus the slow-convergence issue is expected to be resolved.
c) Under the use of constant step size, the regularization parameter λ is also constant in the proposed PDA framework, in contrast to the case of RDA for which the regularization parameter increases linearly in time (see the paragraph under (26)). Due to the constancy of the regularization parameter, the performance of the PDA method is insensitive to the choice of the regularization parameter, as demonstrated in Section IV-A.
d) The PDA method is free from accumulation of the shrinkage effects, as opposed to APFBS and FOBOS (see Section III-D for more details). This remarkable property comes directly from the fact that PDA is based on the dual-averaging framework.

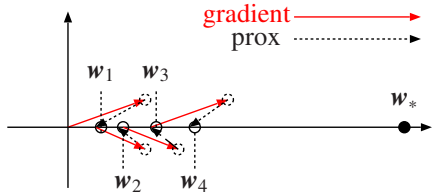
Items a) – c) above are the major differences from RDA, while item d) is common to PDA and RDA. The PDA and RDA frameworks are disjoint basically, as long as the regularization term exists (as long as $\psi_t \neq 0$). In the case of $\psi_t = 0$, PDA for $\boldsymbol{Q}_t := \boldsymbol{I}$ coincides with RDA applied to the squared-distance loss with $\beta_t := 1/\eta$. See Figure 3.

D. Relations to Other Prior Works

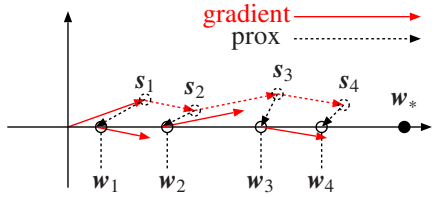
The relations to APFBS, AdaGrad, and the projection-based method are discussed below.

1) APFBS: PDA has a clear advantage in performance over APFBS, as elaborated below. The APFBS iterate [52, 53] (for date-reusing factor 1) is given by $\boldsymbol{w}_t := \mathrm{prox}^{\boldsymbol{Q}_t}_{\eta\psi_t}(\boldsymbol{w}_{t-1} - \eta\boldsymbol{g}_t)$. Figure 4 illustrates the difference between APFBS and PDA. One can see that APFBS performs the proximity operator many times, and its accumulation may cause a serious increase of estimation biases. Thus, APFBS has a tradeoff between the sparsity of the obtained solution and the estimation accuracy, depending on the strength of regularization. In contrast, PDA is free from such an accumulation issue, and hence it can achieve high estimation-accuracy and a high sparsity-level simultaneously. The FOBOS algorithm [9] resembles APFBS, but it considers the ordinary loss functions.

APFBS reduces to the projection-based method (9) in the absence of the regularization term (i.e., $\psi_t := 0$, or equivalently $\lambda := 0$). For the particular choice of C_t in (20) for regression, (9) reproduces the proportionate affine projection algorithm (PAPA) [14, 62]. In the particular case of $r = 1$, it reproduces the (improved) proportionate NLMS (PNLMS) algorithms [12, 13, 63]. If the metric is Euclidean, PAPA and PNLMS reduces to APA and NLMS, respectively. In the classification case, (9) with (23) reproduces the passive aggressive algorithm [64].



(a) APFBS (forward-backward splitting)



(b) PDA (or RDA)

Fig. 4: Illustrations of APFBS and PDA.

TABLE III: Differences among several algorithms.

	Principle	Loss	Metric	Regularizer
AdaGrad-RDA	DA	ordinary loss	ave. grad. (variance $\downarrow$)	ℓ_1 (fixed)
APFBS	SGD	squared distance	proportionate (sparsity $\uparrow$)	weighted ℓ_1
PDA	DA	squared distance	proportionate (sparsity $\uparrow$)	quad.-weighted ℓ_1

2) AdaGrad: AdaGrad [47] is one of the celebrated online learning methods in machine learning, employing a different metric for a different purpose from the one employed by PDA. The idea of AdaGrad is to reduce the variance of the (sub)gradient vectors by summing up the outer-products of the history of the (sub)gradient vectors to build a metric. This metric emphasizes *infrequently occurring* input vectors (or *features* if we borrow the terminology from the original paper [47]). AdaGrad is thus useful when such infrequently occurring input vectors are highly informative and discriminative. As such, although AdaGrad and PDA share the common principle of changing the geometry of the learning space to improve the convergence behaviours, those methods target different situations from each other and one cannot tell which performs better in general. In the experiments presented in the following section, PDA outperforms AdaGrad-RDA and AdaGrad-FOBOS consistently. The AdaGrad method has been applied to the RDA and COMID algorithms [50, 65] with the ordinary loss functions unlike PDA. The relations are summarized in Table III. See Section III-C for the differences between RDA and PDA.

IV. Simulation Results

We show the efficacy of the proposed PDA method in applications to classification and regression. We first show that the proposed method is insensitive to the choice of the regularization parameter (Experiment A); see Section III-C. We then consider two datasets for classification: MNIST hand written digit dataset [66] (Experiment B-1) and RCV text dataset [67] (Experiment B-2). We finally consider three regression problems: sparse system identification (Experiment C-1), acoustic echo cancellation (Experiment C-2), and nonlinear model estimation with multikernel adaptive filtering [68] (Experiment C-3). Whenever numerical instability may happen due to devision-by-zero, regularization is considered with parameter (such as ϵ, δ) 1.0×10^{-5} for all methods throughout the simulations. In each simulation, the parameters for PDA and the other methods compared are chosen, so that the speeds of initial convergence are comparable, within the following ranges: $\lambda \in [1.0 \times 10^{-8}, 10]$, $\eta \in [1.0 \times 10^{-6}, 10]$, $\alpha \in [0, 1]$, $\gamma \in [1, 2]$ (for Experiments A, C-1, and C-2), and $\lambda \in [1.0 \times 10^{-8}, 1.0 \times 10^{6}]$. Although an efficient implementation of the hyperparameter optimization scheme such as mixture-of-experts type approaches [69] could be employed, this experimental section aims to present pure comparisons among the stochastic optimization schemes, and thus an employment of such an elaborate scheme is beyond the scope of the present study.

All the results presented in this section are averages over 300 independent trials. Here, the randomness is with respect to the sparse system, input, and noise for system identification (Experiments A, C-1, and C-3), with respect only to noise for echo cancellation (Experiment C-2), and with respect to the validation sub-dataset for classification (Experiments B-1 and B-2).

A. Experiment A: Sensitivity Analysis

To show that the use of constant step-size and regularization-parameter makes the algorithm insensitive to the choice of λ, we consider the following algorithm:

$$\boldsymbol{w}_t := \arg\min_{\boldsymbol{w}\in\mathbb{R}^n} \left(\left\langle \frac{\eta}{t^b} \boldsymbol{s}_t, \boldsymbol{w} \right\rangle + h(\boldsymbol{w}) + \eta t^a \psi(\boldsymbol{w}) \right), \tag{27}$$

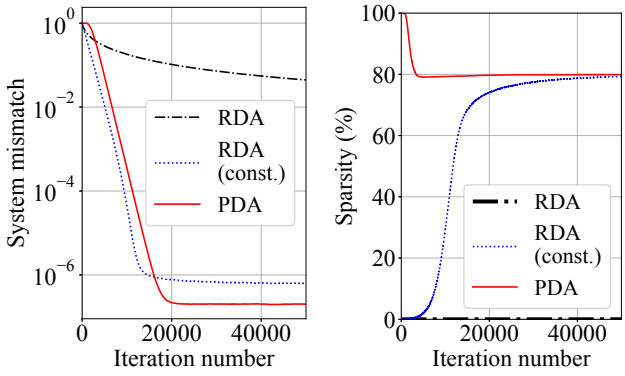
where $a \in \mathbb{R}$ and $b \in \mathbb{R}$ are constants. Letting $a = 1 - b$ in (27) reproduces RDA (26) with $\beta_t := t^b/\eta$. In particular, the case of $a = b = 0.5$ corresponds to $\beta_t := \sqrt{t}/\eta$ (the diminishing step-size case), and the case of $a = 1$, $b = 0$ corresponds to $\beta_t := 1/\eta$ (the constant step-size). Meanwhile, letting $a = b = 0$ in (27) reproduces PDA for $\alpha = 1$ (the case of the ordinary Euclidean metric) which uses the step-size and regularization parameters both constant.

We compare the performances of the above three cases in the online-regression task of identifying a randomly-generated sparse system $\boldsymbol{w}_* \in \mathbb{R}^{1000}$ with sparsity level (proportion of zero components) 80% (see Section II-C for definition of $\boldsymbol{w}_*$). Here, the positions of zero components are chosen randomly with equal probability, and the nonzero components are randomly generated from the i.i.d. uniform distribution $\mathcal{U}[-4, 4]$. The input vector $\boldsymbol{x}_t \in \mathbb{R}^{1000}$ is randomly drawn from the i.i.d. uniform distribution $\mathcal{U}[-2, 2]$, and the noise v_t from the zero-mean i.i.d. normal distribution $\mathcal{N}(0, 0.01)$.

To show the pure effects of using constant step-size and regularization-parameter simultaneously, we employ the ordinary least-square loss function (rather than the squared-distance function), and let $h(\boldsymbol{w}) := \|\boldsymbol{w}\|^2/2$ for all cases. Parameters are summarized in Table IV. Figure 5 plots the learning curves for (a) system mismatch $\|\boldsymbol{w}_* - \boldsymbol{w}_t\|^2/\|\boldsymbol{w}_*\|^2$ and (b) the sparsity. It is seen that RDA with diminishing step size

TABLE IV: Parameters for Experiment A.

Algorithms	λ	η	α
$a = b = 0.5$ (RDA)	0.2	0.008	-
$a = 1$, $b = 0$ (RDA w/ const. step size)	1	0.001	-
$a = b = 0$ (PDA w/ least-square loss)	8000	0.0004	1



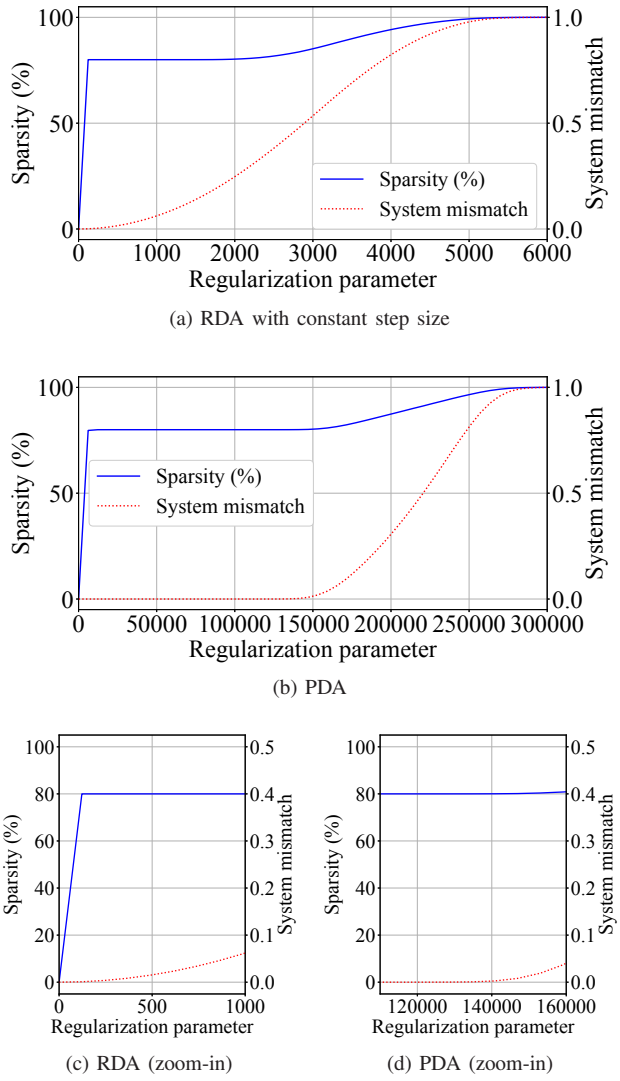
(a) (b)

Fig. 5: Learning curves for best λ in Experiment A. (The sparsity of "RDA" is 0%.)

TABLE V: Parameters for Experiment B-1.

Algorithms	λ	η	α
AdaDelta	-	-	-
AdaGrad-FOBOS	5×10^{-3}	0.1	-
AdaGrad-RDA	10^{-4}	3.4	-
Adam	-	5×10^{-5}	-
RDA	5×10^{-4}	1.5	-
PDA	10^{-4}	0.15	1

performs poor in this simulation setting. We therefore show in Figure 6 the sensitivity curves of PDA and RDA with constant step size to the choice of λ.



(a) RDA with constant step size

(b) PDA

(c) RDA (zoom-in) (d) PDA (zoom-in)

Fig. 6: Sensitivity of RDA and PDA to the choice of λ.

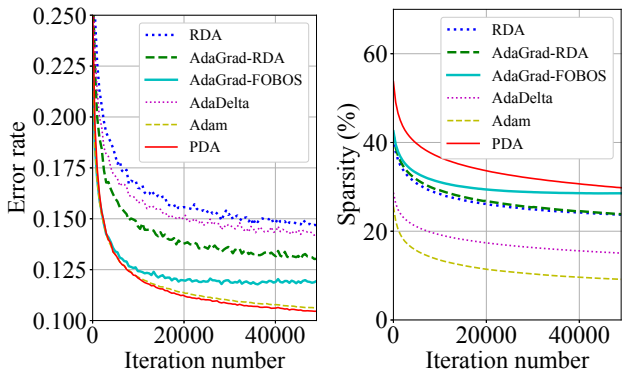
B. Applications to Classification Tasks

The proposed method is compared to RDA [3], AdaGrad-RDA [47], AdaGrad-FOBOS [47], Adam [49], and AdeDelta [35]. PDA employs the halfspace C_t given in (23). All the other algorithms employ the logistic loss

$$\varphi_t(\boldsymbol{w}) = y_t \log\left(1 + e^{-\hat{y}_t}\right) + (1 - y_t) \log\left(\frac{1 + e^{-\hat{y}_t}}{e^{-\hat{y}_t}}\right), \qquad (28)$$

where $\hat{y}_t := \boldsymbol{x}_t^\top \boldsymbol{w}$. For all algorithms, $\psi_t(\boldsymbol{w}) := \lambda \|\boldsymbol{w}\|_1$ is used; i.e., $\alpha = 1$ ($\boldsymbol{Q}_t = \boldsymbol{I}$) for PDA. The dataset is split into a validation sub-dataset (30%) and a training sub-dataset (70%). The error rate, the misclassification ratio for the validation dataset, is adopted as a performance measure. In each trial, the training dataset is shuffled randomly. The one-vs-all method is employed to train the multi-class classifier. Parameters are summarized in Tables V and VI for handwritten digit classification and text classification, respectively.

1) Experiment B-1: Handwritten Digit Classification: MNIST [66] is a handwritten digit dataset. Each datum consists of 28×28 pixels with gray-scale values normalized in the interval $[0, 1]$, and it is labeled by a digit from 0 to 9. The objective is to learn a linear classifier that discriminates the handwritten images. Figure 7 shows the learning curves of



(a) (b)

Fig. 7: Results for Experiment B-1.

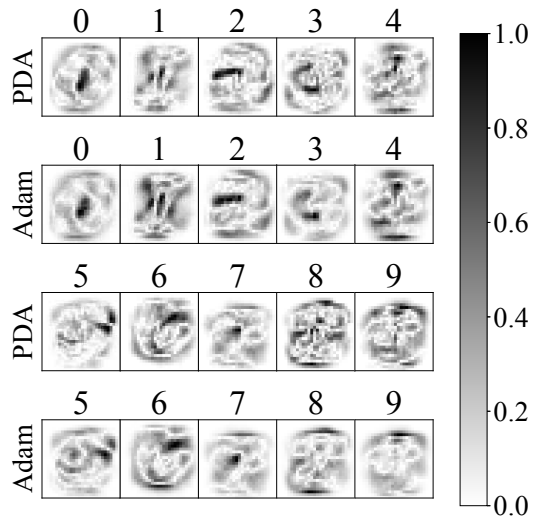
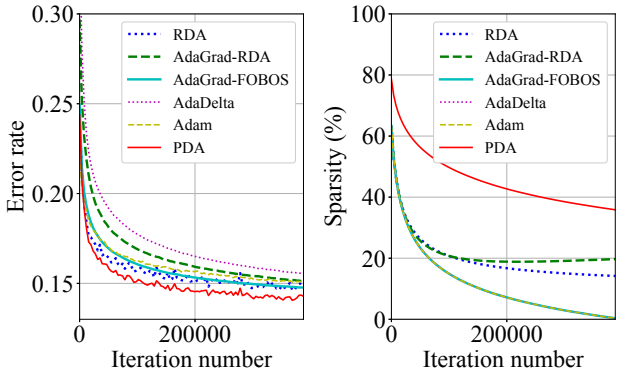


Fig. 8: Visualization of the estimated coefficients for Adam and PDA.



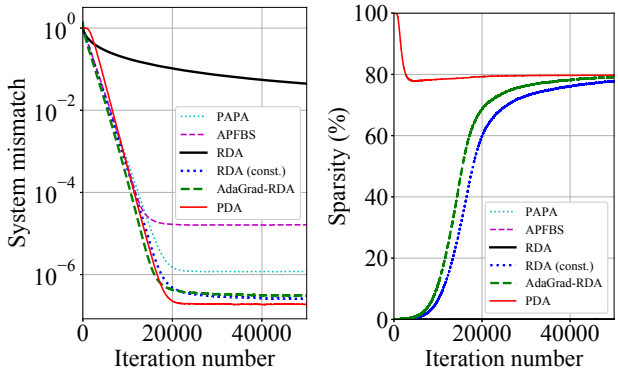
(a) (b)

Fig. 9: Results for Experiment B-2. (The bottom overlapping curves in (b) are "AdaGrad-FOBOS", "AdaDelta", and "Adam".)

TABLE VI: Parameters for Experiment B-2.

Algorithms	λ	η	α
AdaDelta	0.1	10^{-5}	-
AdaGrad-FOBOS	4×10^{-6}	0.1	-
AdaGrad-RDA	10^{-7}	0.1	-
Adam	0.1	0.1	-
RDA	10^{-8}	1.4	-
PDA	0.05	0.3	1

(a) the error rate and (b) the sparsity level for each algorithm. Figure 8 depicts the normalized magnitudes of the components of $\boldsymbol{w}_t$ generated by PDA and Adam, visualizing which parts of the images the classifiers look at to tell whether each given image is a specified number or not. Referring to the two images of number 8 in the figure, for instance, one can see that the classifier obtained by PDA looks at the relevant parts more clearly than the one obtained by Adam.



(a) (b)

Fig. 10: Results for Experiment C-1.

TABLE VII: Parameters for Experiment C-1.

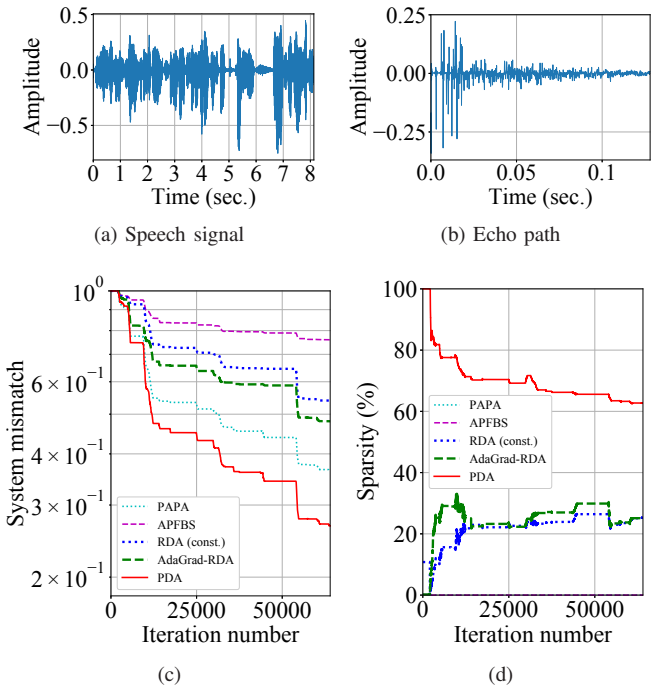
Algorithms	λ	η	α	r
PAPA	-	0.5	0.8	1
APFBS	10^{-5}	0.5	0.8	1
RDA	0.2	0.008	-	-
RDA (constant)	2	0.0004	-	-
AdaGrad-RDA	10^{-3}	1	-	-
PDA	4	0.5	0.8	1

2) Experiment B-2: Text Classification: RCV is a news text dataset [67]. Each datum is based on bag-of-words representations (where the feature vectors contain the frequencies of occurrence of each word) and is associated with some of four labels (Economics, Industrial, Social, and Markets); multiple labels are allowed. The objective is to learn a linear classifier to tell whether a given news-text belongs to a prespecified category. The bag-of-words representations are typically sparse since the number of coefficients is the number of vocabularies that appear in the text data, and each text datum only contains a small fraction of those vocabularies. Figure 9 shows the results.

C. Applications to Regression Tasks

For the linear regression tasks (Experiments C-1 and C-2), the proposed method is compared to PAPA [14, 62], APFBS [52, 53], RDA [3], and AdaGrad-RDA [47]. For RDA and AdaGrad-RDA, $\varphi_t^{\mathrm{LS}}(\boldsymbol{w})$ in (10) and $\psi_t(\boldsymbol{w}) := \lambda \|\boldsymbol{w}\|_1$ are used. For the other algorithms, $\varphi_t(\boldsymbol{w})$ in (7) is used with the metric in (14) for fairness. Also for fairness, the quadratically-weighted ℓ_1 norm in (17) is used for both PDA and APFBS. For the nonlinear regression task (Experiment C-3), the proposed method (the *nonlinear* version of the PDA method) is compared to the existing multikernel adaptive filtering algorithm based on APFBS [70]. The system mismatch is used as a performance measure for sparse system identification and echo cancellation, and MSE is used for nonlinear model estimation. Parameters are summarized in Tables VII and VIII for sparse system identification and echo cancellation, respectively.

1) Experiment C-1: Sparse System Identification: The sparse system considered here is the same as used in Section IV-A. Figure 10 depicts the results. For the RDA algorithm,



(a) Speech signal (b) Echo path

(c) (d)

Fig. 11: Results for Experiment C-2.

we test both diminishing and constant step-size sequences; "const." in the figure indicates constant step-size. Note that PAPA, APFBS, and RDA (with diminishing step-size) generate the coefficient vectors with sparsity level 0% in this case. (PAPA uses no sparsity-promoting regularizer, and hence it always gives sparsity level 0% in principle.)

2) Experiment C-2: Acoustic Echo Cancellation: Acoustic echo paths tend to decay in time and thus are often assumed *weakly sparse* [12, 13, 15, 63]; i.e., many of the coefficients are (not exactly but) nearly zero. Figures 11a, 11b depicts the speech signal and the echo path used in the simulation. The sampling frequency of speech signal and echo path is 8.0 kHz. The learning is stopped whenever the amplitude of input signal is below the threshold 1.0×10^{-4} for avoiding divergence. The noise is zero-mean i.i.d. Gaussian with the signal to noise ratio (SNR) 20 dB. Figures 11c and 11d show the results. For the RDA algorithm, the results for the diminishing step size are poor (as in the case of Experiment C-1) and are hence omitted.

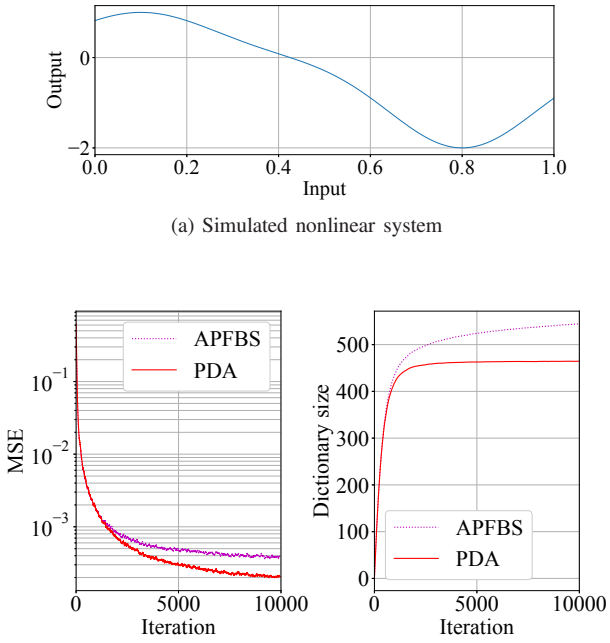
3) Experiment C-3: Nonlinear model estimation: A nonlinear extension of PDA is possible based on the multikernel adaptive filtering framework [68, 71, 72], as elaborated in Appendix A. The extended PDA is finally applied to the problem of estimating the following nonlinear system [70] (see Figure 12a):

$$y_t := \exp\left(-20(x_t-0.1)^2\right) - 2\exp\left(-20(x_t-0.8)^2\right) + u_t, \quad (29)$$

where $x_t \sim \mathcal{U}[0,1]$ and $u_t \sim \mathcal{N}(0, 0.01)$. Parameter settings are given in Appendix A. Figure 12 plots (a) the MSE learning curves and (b) the evolutions of dictionary size. Here, a small dictionary size implies that the estimated model is sparse.

TABLE VIII: Parameters for Experiment C-2.

Algorithms	λ	η	α	r
PAPA	-	0.1	0.2	2
APFBS	10^{-5}	0.2	0.01	2
RDA (constant)	10^{-2}	0.02	-	-
AdaGrad-RDA	10^{-5}	0.02	-	-
PDA	0.05	0.2	0.2	2



(a) Simulated nonlinear system

(b) (c)

Fig. 12: Results for Experiment C-3.

D. Discussions

Through the whole simulations, PDA achieves the sparsest solutions among all the methods, and this leads to the best classification/estimation accuracy as well as the efficiency in terms of the validation-data classification/regression costs. Let us present some remarks on each simulation.

- Experiment A (sensitivity analysis): First, PDA is insensitive to the choice of λ in contrast to the high sensitivity of RDA (with constant step size). Indeed, the system mismatch of PDA stays constant with the correct sparsity level 80% for a wide range of λ, while the system mismatch of RDA starts increasing as soon as the sparsity level reaches the correct one. This notable stability is an important advantage of the proposed method. Second, the constant step-size case ($a = 1$, $b = 0$) gives much higher accuracy than the diminishing step-size case ($a = 0.5$, $b = 0.5$) in the RDA framework. This is due to the smoothness of the loss function.
- Experiment B-1 (MNIST hand-written-digit classification): Although Adam achieves low error rates comparable to PDA, its sparsity 9.2% is the lowest among all the methods (i.e., it yields the densest $\boldsymbol{w}_t$).
- Experiment B-2 (RCV news-text classification): PDA successfully extracts the sparse structure of the text dataset,

thereby gaining the high classification-accuracy.

- Experiment C-1 (sparse system identification): It achieves a sparsity level close to the true level 80% approximately in 5,000 iterations, whereas PAPA, APFBS, and RDA generate dense coefficient vectors of sparsity level nearly 0% and AdaGrad-RDA takes about 40,000 iterations to achieve sparsity level 40% (which is a half of the true one). (Recall here the difference between APFBS and RDA discussed in Section III-D.)
- Experiment C-2 (echo cancellation): Although the echo path is sparse only in the weak sense, PDA still achieves a sparser solution and attains better performances than the other methods. The improvements come from the better bias-variance tradeoff; i.e., the proximal (shrinkage) step vanishes the nearly-zero coefficients, and this often leads to significant reduction of estimation variance at the price of a slight increase of bias. (The same applies to Experiment B-1.)
- Experiment C-3 (nonlinear model estimation): PDA achieves lower MSEs and a smaller dictionary-size (the number of basis functions used for estimation) than APFBS.

In summary, all the results support the clear benefits from using the squared-distance loss (the projection), rather than the ordinary losses. In particular, in the applications to classification, the standard Euclidean metric was employed because the use of the sparsity-promoting metric yielded no significant gains in this specific application. The reason would be because faster convergence to an optimal solution does not directly affect the performance measure (i.e., misclassification ratio) of classification.[5] The gains of PDA in the classification applications come solely from the use of projection as well as the fixed step-size and regularization parameters. The gains from the sparsity-promoting metric and the quadratically-weighted ℓ_1 regularizer are significant in the linear-regression applications. Through the experiments, PDA turns out to enjoy

1) the considerable insensitivity to the choice of the regularization parameter λ, and
2) the remarkable sparsity-seeking property which has a striking difference from those of the existing methods (including RDA and APFBS/FOBOS).

We finally mention that, if other prior knowledge than sparseness is available, one may accommodate such prior knowledge as convex constraints/penalties based on convex analysis [73, 74] and may consider to use it within the PDA framework. One may also accommodate it in the prior distribution of observed data based on the statistical Bayesian framework (cf., e.g., [75, 76]). Further discussions in this direction are out of the scope of this paper.

V. Conclusion

We proposed the efficient regularized stochastic optimization framework named PDA which enjoys high estimation-accuracy and excellent sparsity-promoting capability (and hence enjoys low computational costs for validation-data evaluations as well). In the PDA framework, classification/regression tasks involving sparsity are cast as the stochastic minimization problem of the smooth loss function (i.e., the squared-distance function) penalized by the sparsity-promoting regularizer. The proposed method was derived based on RDA with the following three ideas: (i) the use of projection (a direct consequence of the use of the squared-distance loss), (ii) the simultaneous use of the sparsity-promoting metric and the quadratically-weighted ℓ_1 regularizer, and (iii) the use of step-size and regularization parameter both constant. The use of the quadratically-weighted ℓ_1 regularizer successfully reduces the estimation variance while alleviating those possible large biases caused by the use of the sparsity-promoting metric. To the best of authors' knowledge, the squared-distance function and the sparsity-promoting metric have not yet been studied under the dual averaging framework. The proposed method is insensitive to the choice of the regularization parameter. Numerical examples showed the advantages of the proposed method over the existing methods including AdaGrad and APFBS in its applications to regression and classification with real data (as well as some synthetic data).

Appendix A
A Nonlinear Extension Based on Multikernel Adaptive Filter

We consider the following nonlinear system model:

$$y_t = f(\boldsymbol{x}_t) + \nu_t, \quad \boldsymbol{x}_t \in \mathbb{R}^n, \quad \nu_t \in \mathbb{R}, \tag{30}$$

where the function $f : \mathbb{R}^n \to \mathbb{R}$ is usually assumed smooth. Multikernel adaptive filtering [68, 71] is based on the following model:

$$f_t(\boldsymbol{x}) := \sum_{m\in\mathcal{M}} \underbrace{\sum_{j\in\mathcal{J}_t} h_{j,t}^{(m)} k_m(\boldsymbol{x}, \boldsymbol{x}_j)}_{m\text{th model}}, \quad h_{j,t}^{(m)} \in \mathbb{R}, \tag{31}$$

where $k_m : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, $m \in \mathcal{M} := \{1, 2, \cdots, M\}$, are the positive definite kernels to be used, and $\{k_m(\cdot, \boldsymbol{x}_j)\}_{m\in\mathcal{M}, j\in\mathcal{J}_t}$ is the dictionary of size r_t with the index set $\mathcal{J}_t := \{j_1^{(t)}, j_2^{(t)}, \cdots, j_{r_t}^{(t)}\} \subset \{1, 2, \cdots, t\}$. The model in (31) can be rewritten as

$$f_t(\boldsymbol{x}_t) = \langle \boldsymbol{H}_t, \boldsymbol{K}_t \rangle_{\mathrm{F}} := \mathrm{trace}(\boldsymbol{H}_t^{\mathsf{T}} \boldsymbol{K}_t), \tag{32}$$

where

$$\begin{aligned}
\boldsymbol{H}_t &:= \left[\boldsymbol{h}_{j_1^{(t)},t}\ \boldsymbol{h}_{j_2^{(t)},t}\ \cdots\ \boldsymbol{h}_{j_{r_t}^{(t)},t}\right] \in \mathbb{R}^{M\times r_t},\\
\boldsymbol{h}_{j,t} &:= \left[h_{j,t}^{(1)}, h_{j,t}^{(2)}, \cdots, h_{j,t}^{(M)}\right]^{\mathsf{T}} \in \mathbb{R}^M,\\
\boldsymbol{K}_t &:= \left[\boldsymbol{k}_{j_1^{(t)},t}\ \boldsymbol{k}_{j_2^{(t)},t}\ \cdots\ \boldsymbol{k}_{j_{r_t}^{(t)},t}\right] \in \mathbb{R}^{M\times r_t},\\
\boldsymbol{k}_{j,t} &:= \left[k_1(\boldsymbol{x}_t, \boldsymbol{x}_j), k_2(\boldsymbol{x}_t, \boldsymbol{x}_j), \cdots, k_M(\boldsymbol{x}_t, \boldsymbol{x}_j)\right]^{\mathsf{T}} \in \mathbb{R}^M.
\end{aligned}$$

Define the loss function $l_t(\boldsymbol{H}) := \varphi_t(\boldsymbol{H}) + \psi_t(\boldsymbol{H})$, $\boldsymbol{H} \in \mathbb{R}^{M\times r_t}$, with

$$\varphi_t(\boldsymbol{H}) := \frac{1}{2} d^2(\boldsymbol{H}, C_t), \quad \psi_t(\boldsymbol{H}) = \lambda \sum_{j\in\mathcal{J}_t} \left\|\boldsymbol{h}_j\right\|, \tag{33}$$

[5] The intersection of the halfspaces C_t given in (23) is unbounded (if exists), because $\boldsymbol{w} \in C_t$ implies $\alpha\boldsymbol{w} \in C_t$ for any $\alpha \geq 1$, as well as convex.

TABLE IX: Parameters for Experiment C-3.

Algorithms	η	λ	$\epsilon_{\mathcal{J}}$	$r_{\max}$
APFBS	0.3	10^{-8}	10^{-12}	20
PDA	0.3	10^{-6}	10^{-12}	20

where $d(\boldsymbol{H}, C_t) := \min_{\boldsymbol{Y}\in C_t} \|\boldsymbol{H}-\boldsymbol{Y}\|_{\mathrm{F}} := \sqrt{\langle \boldsymbol{H}-\boldsymbol{Y}, \boldsymbol{H}-\boldsymbol{Y}\rangle_{\mathrm{F}}}$ with $C_t := \{\boldsymbol{H} \in \mathbb{R}^{M\times r_t} \mid \langle \boldsymbol{H}, \boldsymbol{K}_t\rangle_{\mathrm{F}} = y_t\}$. The (ordinary) gradient of $\varphi_t(\boldsymbol{H})$ is given by $\nabla\varphi_t(\boldsymbol{H}) = \boldsymbol{H} - P_{C_t}(\boldsymbol{H})$ with

$$P_{C_t}(\boldsymbol{H}) : = \arg\min_{\boldsymbol{Y}\in C_t} \|\boldsymbol{H}-\boldsymbol{Y}\|_{\mathrm{F}}^2 = \boldsymbol{H} - \frac{\langle \boldsymbol{H}, \boldsymbol{K}_t\rangle_{\mathrm{F}} - y_t}{\|\boldsymbol{K}_t\|_{\mathrm{F}}^2}\boldsymbol{K}_t. \quad (34)$$

The PDA update for multikernel adaptive filtering is given as

$$\hat{\boldsymbol{H}}_t := -\eta \sum_{\tau=1}^{t} \nabla\varphi_\tau(\boldsymbol{H}_{\tau-1}), \quad (35)$$

$$\begin{aligned}\boldsymbol{H}_t &:= \mathrm{prox}_{\psi_t}\left(\hat{\boldsymbol{H}}_t\right)\\ &= \sum_{j\in\mathcal{J}_t} \max\left\{1 - \frac{\lambda}{\left\|\hat{\boldsymbol{h}}_{j,t}\right\|}, 0\right\} \hat{\boldsymbol{h}}_{j,t}\boldsymbol{e}_{j,r_n}^{\mathsf{T}}, \end{aligned} \quad (36)$$

where $\hat{\boldsymbol{h}}_{j_i^{(t)},t}$ is the ith column of $\hat{\boldsymbol{H}}_t$, and $\boldsymbol{e}_{j,r_n}$ is the length-r_n unit vector that has one at the jth entry and zeros elsewhere. The dictionary is constructed in an online fashion as follows [70]:

Dictionary update

Iteration : For $t = 1, 2, 3, \cdots$

1. Grow the dictionary as $\mathcal{J}_{t-1}^{+} := \mathcal{J}_{t-1} \cup \{t\}$
2. Update the coefficients by (35) and (36)
3. If $r_t \geq r_{\max}$, refine the dictionary as $\mathcal{J}_t := \{j \in \mathcal{J}_{t-1}^{+} \mid \left\|\boldsymbol{h}_{j,t}\right\| \geq \epsilon_{\mathcal{J}}\}$ for some small $\epsilon_{\mathcal{J}} > 0$

In the simulation presented in Section IV-C3, we adopt the normalized Gaussian kernel [77]

$$k_m(\boldsymbol{x}, \boldsymbol{z}) := \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|^2}{2\sigma_m^2}\right), \quad \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n, \quad (37)$$

where $\sigma_m^2 := a \times 10^b$ is the kernel parameter, where $a \in \{1, 2, \cdots, 9\}$ and $b \in \{-4, -3, \cdots, 1\}$ ($M = 54$). The other parameters are summarized in Table IX.

References

[1] A. Ushio and M. Yukawa, "Projection-based dual averaging for stochastic sparse optimization," in *Proc. IEEE ICASSP*, 2017, pp. 2307–2311.

[2] Y. Nesterov, "Primal-dual subgradient methods for convex problems," *Mathematical Programming*, vol. 120, no. 1, pp. 221–259, Aug. 2009.

[3] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," *Journal of Machine Learning Research*, vol. 11, pp. 2543–2596, Oct. 2010.

[4] I. Yamada and N. Ogura, "Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions," *Numer. Funct. Anal. Optim.*, vol. 25, no. 7&8, pp. 593–617, 2004.

[5] K. Slavakis, I. Yamada, and N. Ogura, "The adaptive projected subgradient method over the fixed point set of strongly attracting nonexpansive mappings," *Numerical Functional Analysis and Optimization*, vol. 27, no. 7-8, pp. 905–930, Dec. 2006.

[6] S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections: a unifying framework for linear and nonlinear classification and regression tasks," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, Jan. 2011.

[7] H. Stark and Y. Yang, *Vector Space Projections—A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics*. New York: John Wiley & Sons, 1998.

[8] A. Kalai and S. Vempala, "Efficient algorithms for online decision problems," *Journal of Computer and System Sciences*, vol. 71, no. 3, pp. 291–307, Oct. 2005.

[9] Y. Singer and J. C. Duchi, "Efficient learning using forward-backward splitting," in *Advances in Neural Information Processing Systems*, Dec. 2009, pp. 495–503.

[10] S. Makino and Y. Kaneda, "Exponentially weighted step-size projection algorithm for acoustic echo cancellers," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, vol. 75, no. 11, pp. 1500–1508, Nov. 1992.

[11] S. Makino, Y. Kaneda, and N. Koizumi, "Exponentially weighted stepsize NLMS adaptive filter based on the statistics of a room impulse response," *IEEE Trans. Speech and Audio Processing*, vol. 1, no. 1, pp. 101–108, Jan. 1993.

[12] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 5, pp. 508–518, Sep. 2000.

[13] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, vol. 1, 1998, pp. 394–398.

[14] J. Benesty, Y. A. Huang, J. Chen, and P. A. Naylor, "Adaptive algorithms for the identification of sparse impulse responses," *Selected Methods for Acoustic Echo and Noise Control*, vol. 5, pp. 125–153, 2006.

[15] M. Yukawa, K. Slavakis, and I. Yamada, "Adaptive parallel quadratic-metric projection algorithms," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1665–1680, Jul. 2007.

[16] M. Yukawa and I. Yamada, "A unified view of adaptive variable-metric projection algorithms," *EURASIP J. Advances in Signal Processing*, vol. 2009, Article ID 589260, 13 pages, no. 1, 2009.

[17] J.-I. Nagumo and A. Noda, "A learning method for system identification," *IEEE Trans. Automatic Control*, vol. 12, no. 3, pp. 282–287, Jun. 1967.

[18] A. E. Albert and L. S. Gardner Jr., *Stochastic Approximation and Nonlinear Regression*. Cambridge MA: MIT Press, 1967.

[19] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electronics and Communications in Japan (Part I: Communications)*, vol. 67, no. 5, pp. 19–27, May. 1984.

[20] T. Hinamoto and S. Maekawa, "Extended theory of learning identification," *Electrical Engineering in Japan*, vol. 95, no. 5, pp. 101–107, Oct. 1975.

[21] M. Yukawa, K. Slavakis, and I. Yamada, "Multi-domain adaptive learning based on feasibility splitting and adaptive projected subgradient method," *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, vol. 93, no. 2, pp. 456–466, Feb. 2010.

[22] M. Yukawa, R. L. G. Cavalcante, and I. Yamada, "Efficient blind MAI suppression in DS/CDMA systems by embedded constraint parallel projection techniques," *IEICE Trans. Fundamentals*, vol. E88-A, no. 8, pp. 2062–2071, Aug. 2005.

[23] S. Narayan, A. Peterson, and M. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 31, no. 3, pp. 609–615, Jun. 1983.

[24] D. F. Marshall, W. K. Jenkins, and J. Murphy, "The use of orthogonal transforms for improving performance of adaptive filters," *IEEE Trans. Circuits and Systems*, vol. 36, no. 4, pp. 474–484, Apr. 1989.

[25] F. Beaufays, "Transform-domain adaptive filters: An analytical approach," *IEEE Trans. Signal Processing*, vol. 43, no. 2, pp. 422–431, Feb. 1995.

[26] B. Widrow and S. D. Stearns, "Adaptive signal processing," *Englewood Cliffs, NJ, Prentice-Hall, Inc., 1985.*, vol. 1, 1985.

[27] P. S. Diniz, M. L. de Campos, and A. Antoniou, "Analysis of LMS-Newton adaptive filtering algorithms with variable convergence factor," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 617–627, Mar. 1995.

[28] B. Farhang-Boroujeny, *Adaptive filters: theory and applications*. John Wiley & Sons, 2013.

[29] D. F. Marshall and W. K. Jenkins, "A fast quasi-Newton adaptive filtering algorithm," *IEEE Trans. Signal Processing*, vol. 40, no. 7, pp. 1652–1662, Jul. 1992.

[30] M. L. De Campos and A. Antoniou, "A new quasi-Newton adaptive filtering algorithm," *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, vol. 44, no. 11, pp. 924–934, Nov. 1997.

[31] M. Yukawa, "Krylov-proportionate adaptive filtering techniques not limited to sparse systems," *IEEE Trans. Signal Processing*, vol. 57, no. 3, pp. 927–943, Mar. 2009.

[32] M. Yukawa and W. Utschick, "Proportionate adaptive algorithm for non-sparse systems based on krylov subspace and constrained optimization," in *Proc. IEEE ICASSP*. IEEE, 2009, pp. 3121–3124.

[33] ——, "A fast stochastic gradient algorithm: maximal use of sparsification benefits under computational constraints," *IEICE Trans. Fundamentals of Electronics, Communications and Computer sciences*, vol. 93, no. 2, pp. 467–475, Feb. 2010.
[34] A. Bordes, L. Bottou, and P. Gallinari, "SGD-QN: Careful quasi-newton stochastic gradient descent," *Journal of Machine Learning Research*, vol. 10, pp. 1737–1754, Jul. 2009.
[35] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.
[36] T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learning rates," in *Proc. International Conference on Machine Learning*, 2013, pp. 343–351.
[37] T. Schaul and Y. LeCun, "Adaptive learning rates and parallelization for stochastic, sparse, non-smooth gradients," in *Proc. International Conference on Learning Representations*, Scottsdale, AZ, 2013.
[38] O. Vinyals and D. Povey, "Krylov subspace descent for deep learning," in *Artificial Intelligence and Statistics*, Mar. 2012, pp. 1261–1268.
[39] N. N. Schraudolph, "Fast curvature matrix-vector products for second-order gradient descent," *Neural computation*, vol. 14, no. 7, pp. 1723–1738, Jul. 2002.
[40] J. Martens, "Deep learning via hessian-free optimization," in *Proc. International Conference on Machine Learning*, 2010, pp. 735–742.
[41] R. H. Byrd, S. L. Hansen, J. Nocedal, and Y. Singer, "A stochastic quasi-Newton method for large-scale optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1008–1031, Apr. 2016.
[42] N. N. Schraudolph, J. Yu, and S. Günter, "A stochastic quasi-Newton method for online convex optimization," in *Artificial Intelligence and Statistics*, Mar. 2007, pp. 436–443.
[43] A. Mokhtari and A. Ribeiro, "RES: Regularized stochastic BFGS algorithm," *IEEE Trans. Signal Processing*, vol. 62, no. 23, pp. 6089–6104, Dec. 2014.
[44] S. Amari, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, no. 2, pp. 251–276, Feb. 1998.
[45] N. L. Roux, P.-A. Manzagol, and Y. Bengio, "Topmoumoute online natural gradient algorithm," in *Proc. Advances in Neural Information Processing Systems*, 2008, pp. 849–856.
[46] N. L. Roux and A. W. Fitzgibbon, "A fast natural Newton method," in *Proc. International Conference on Machine Learning*, 2010, pp. 623–630.
[47] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, Jul. 2011.
[48] T. Tieleman and G. Hinton, "Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural Networks for Machine Learning*, vol. 4, no. 2, pp. 26–31, 2012.
[49] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conference on Learning and Representations*, 2015, pp. 1–13.
[50] J. C. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari, "Composite objective mirror descent." in *Proc. COLT*, 2010, pp. 14–26.
[51] J. Kivinen and M. K. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Information and Computation*, vol. 132, no. 1, pp. 1–63, Jan. 1997.
[52] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. IEEE ICASSP*, 2010, pp. 3734–3737.
[53] M. Yamagishi, M. Yukawa, and I. Yamada, "Acceleration of adaptive proximal forward-backward splitting method and its application to sparse system identification," in *Proc. IEEE ICASSP*, 2011, pp. 4296–4299.
[54] B. Widrow and M. E. Hoff, "Adaptive switching circuits," in *IRE WESCON convention record*, vol. 4, no. 1. New York, Jun. 1960, pp. 96–104.
[55] M. Yukawa and I. Yamada, "Two product-space formulations for unifying multiple metrics in set-theoretic adaptive filtering," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2010, pp. 1010–1014.
[56] O. Toda, M. Yukawa, S. Sasaki, and H. Kikuchi, "An efficient adaptive filtering scheme based on combining multiple metrics," *IEICE Trans. Fundamentals*, vol. E97-A, no. 3, pp. 800–808, Mar. 2014.
[57] J. Benesty and S. L. Gay, "An improved pnlms algorithm," in *Proc. IEEE ICASSP*, vol. 2, May. 2002, pp. II–1881.
[58] M. Yukawa, Y. Tawara, M. Yamagishi, and I. Yamada, "Sparsity-aware adaptive filters based on ℓ_p-norm inspired soft-thresholding technique," in *Proc. IEEE ISCAS*, 2012, pp. 2749–2752.
[59] M. Yamagishi, M. Yukawa, and I. Yamada, "Shrinkage tuning based on an unbiased MSE estimate for sparsity-aware adaptive filtering," in *Proc. IEEE ICASSP*, 2014, pp. 5514–5518.
[60] ——, "Automatic shrinkage tuning based on a system-mismatch estimate for sparsity-aware adaptive filtering," in *Proc. IEEE ICASSP*, 2017, pp. 4800–4804.
[61] K. Jeong, M. Yukawa, M. Yamagishi, and I. Yamada, "Automatic shrinkage tuning robust to input correlation for sparsity-aware adaptive filtering," in *Proc. IEEE ICASSP*, 2018, pp. 4314–4318.
[62] T. Gansler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 6, pp. 656–663, Nov. 2000.
[63] C. Paleologu, J. Benesty, and S. Ciochin, "An improved proportionate NLMS algorithm based on the ℓ_0 norm," in *Proc. IEEE ICASSP*, 2010, pp. 309–312.
[64] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *Journal of Machine Learning Research*, vol. 7, pp. 551–585, Mar. 2006.
[65] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," 2008, http://www.math.washington.edu/tseng/papers/apgm.pdf.
[66] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[67] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "RCV1: A new benchmark collection for text categorization research," *Journal of Machine Learning Research*, vol. 5, pp. 361–397, Apr. 2004.
[68] M. Yukawa, "Multikernel adaptive filtering," *IEEE Trans. Signal Processing*, vol. 60, no. 9, pp. 4672–4682, Sept. 2012.
[69] K. Gokcesu and S. S. Kozat, "Online anomaly detection with minimax optimal density estimation in nonstationary environments," *IEEE Trans. Signal Process.*, vol. 66, no. 5, pp. 1213–1227, Mar. 2018.
[70] M. Yukawa and R. Ishii, "Online model selection and learning by multikernel adaptive filtering," in *Proc. EUSIPCO*, 2013, pp. 1–5.
[71] ——, "On adaptivity of online model selection method based on multikernel adaptive filtering," in *Proc. APSIPA-ASC*, 2013, pp. 1–6.
[72] M. Yukawa, "Adaptive learning in Cartesian product of reproducing kernel Hilbert spaces," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6037–6048, Nov. 2015.
[73] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, 1st ed. New York: NY: Springer, 2011.
[74] I. Yamada, M. Yukawa, and M. Yamagishi, "Minimizing the Moreau envelope of nonsmooth convex functions over the fixed point set of certain quasi-nonexpansive mappings," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, ser. Optimization and Its Applications, vol. 49. New York: Springer, 2011, pp. 345–390.
[75] N. Akhtar and A. Mian, "Nonparametric, coupled, Bayesian, dictionary, and classifier learning for hyperspectral classification," *IEEE Trans. Neural Networks and Learning Systems*, vol. 29, no. 9, pp. 4038–4050, Sept. 2018.
[76] S. Liu, J. Jia, Y. D. Zhang, and Y. Yang, "Image reconstruction in electrical impedance tomography based on structure-aware sparse Bayesian learning," *IEEE Trans. Medical Imaging*, vol. 37, no. 9, pp. 2090–2102, Sept. 2018.
[77] O. Toda and M. Yukawa, "On kernel design for online model selection by Gaussian multikernel adaptive filtering," in *Proc. APSIPA Annual Summit and Conference*, 2014, pp. 1–5.

Asahi Ushio (S'16) received the B.E. and the M.E. degrees from Keio University in 2016 and 2018, respectively. He is currently working for Cogent Labs, Japan. His research interests include machine learning, sparse optimization, and signal processing.

Masahiro Yukawa (S'05–M'06) received the B.E., M.E., and Ph.D. degrees from the Tokyo Institute of Technology, Tokyo, Japan, in 2002, 2004, and 2006, respectively. He was Visiting Researcher/Professor with the University of York, U.K. (Oct. 2006–Mar. 2007), with the Technical University of Munich, Germany (July 2008–Nov. 2008), and with the Technical University of Berlin, Germany (Apr. 2016–Feb. 2017). He was Special Postdoctoral Researcher with RIKEN, Japan (2007–2010), and Associate Professor with Niigata University, Japan (2010–2013). He is currently Associate Professor with the Department of Electronics and Electrical Engineering, Keio University, Yokohama, Japan. He has been appointed as Visiting Scientist with the AIP Center, RIKEN, Japan, since July 2017. His research interests include mathematical adaptive signal processing, convex/sparse optimization, and machine learning.

Dr. Yukawa was a recipient of the Research Fellowship of the Japan Society for the Promotion of Science from April 2005 to March 2007. He received the Excellent Paper Award and the Young Researcher Award from the IEICE in 2006 and in 2010, respectively, the Yasujiro Niwa Outstanding Paper Award in 2007, the Ericsson Young Scientist Award in 2009, the TELECOM System Technology Award in 2014, the Young Scientists' Prize, the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology in 2014, the KDDI Foundation Research Award in 2015, and the FFIT Academic Award in 2016. He served as Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING (2015–2019), Multidimensional Systems and Signal Processing (2012–2016), and the IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences (2009–2013). He is a member of the Institute of Electronics, Information and Communication Engineers.