A SPARSE SYSTEM IDENTIFICATION BY USING ADAPTIVELY-WEIGHTED TOTAL VARIATION VIA A PRIMAL-DUAL SPLITTING APPROACH

Shunsuke Ono, Masao Yamagishi, and Isao Yamada

Department of Communications and Computer Engineering, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8550, Japan

ABSTRACT

Observing that sparse systems are almost smooth, we propose to utilize the newly-introduced *adaptively-weighted total variation* (AWTV) for sparse system identification. In our formulation, a sparse system identification problem is posed as a sequential suppression of a time-varying cost function: the sum of AWTV and a data-fidelity term. In order to handle such a non-differentiable cost function efficiently, we propose a time-varying extension of a primal-dual splitting type algorithm, named the *adaptive primal-dual splitting method* (APDS). APDS is free from operator inversion or other highly complex operations, resulting in computationally efficient implementation in online manner. Moreover, APDS realizes that the sequence defined in a certain product space monotonically approaches the solution set of the current cost function, i.e., the sequence generated by APDS pursues desired replicas of the unknown system in each time-step. Our scheme is applied to a network echo cancellation problem where it shows excellent performance compared with conventional methods.

Index Terms— adaptive filtering, sparse system identification, total variation, primal-dual splitting

1. INTRODUCTION

Sparse system identification, i.e., the system to be estimated is assumed to be sparse, arises in many applications including network/acoustic echo cancellation and channel estimation/equalization. For estimating such an unknown sparse system efficiently, adaptive filtering methods using ℓ^0 pseudonorm/ℓ^1 norm/their variants as a sparsity-inducing term have been developed [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. The *adaptive proximal forward-backward splitting method* (APFBS) [2, 7, 9, 10] and *adaptive Douglas-Rachford splitting method* (ADRS) [8] are proximal adaptive filtering methods that can handle a cost function employing the *adaptively-weighted* ℓ_1 *norm* (AWℓ_1) known as a powerful sparsity-inducing term, and they can suppress such a non-differentiable cost function with reasonably-low computational convexity by using the notion of the proximity operator (see the footnote 3). Indeed, they have achieved excellent performance in sparse system identification.

Incidentally, as observed in the left of Fig. 1, many sparse systems can be seen to be smooth (inactive coefficients) with few sharp edges (active coefficients). Moreover, since the notion of smoothness takes a relative information between neighbouring coefficients into account, promoting smoothness is expected to result in a better convergence property than the case where the information of the coefficients are treated independently like ℓ^0/ℓ^1 cases. These observations motivate us to utilize the total variation [11], known as a powerful edge-keeping smoother in image processing, for sparse system identification.

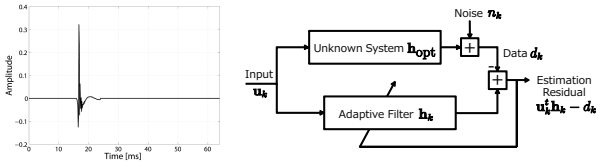

Fig. 1. Sparse system and adaptive filtering strategy.

The first contribution of this paper is to propose an adaptive extension of the so-called *total variation* [11], for sparse system identification. We name it the *adaptively-weighted total variation* (AWTV). AWTV is defined as the sum of the adaptively-weighted absolute differences of the filter coefficients. (for the details of the weight controlling, see Section 3.1), so that we can efficiently promote the smoothness in online manner by suppressing AWTV.

Different from the case of AWℓ_1, it is hard to suppress cost functions employing AWTV using conventional adaptive filtering methods due to the composition of a discrete gradient operator. ADRS is the only existing method that can deal with AWTV via a certain splitting technique. In this case, however, ADRS requires operator inversion in each time-step whose computational cost is usually not accepted in adaptive strategy.

The second contribution of this paper is to propose a novel proximal adaptive filtering method to overcome the above-mentioned difficulty. Our proposed method is a natural time-varying extension of the primal-dual splitting method [12], which is one of primal-dual splitting type algorithms and has been applied to image processing [13], and thus we call the proposed method the *adaptive primal-dual splitting method* (APDS). APDS is superior to existing adaptive methods in terms of the treatment of non-differentiable convex functions involving linear operators like AWTV because it can suppress cost functions employing such a function without using any computationally expensive procedure. Moreover, APDS has an attractive property, that is, the sequence generated by APDS in each time-step, which corresponds to the pair of the current estimate and its dual, monotonically approaches the solution set of the current cost function defined in the product space of primal and dual domain. In other words, the sequence pursues a time-varying set that is expected to contain the unknown system. APDS with AWTV is applied to a network echo cancellation where it shows excellent performance compared to existing adaptive filtering methods.

We thank the reviewers for their careful reading and valuable comments. This work is supported in part by JSPS Grants-in-Aid for JSPS fellows (24·2522), for Research Activity start-up (24800022), and (B-21300091).

2. SPARSE SYSTEM IDENTIFICATION PROBLEM

Let $\mathbb{R}$, $\mathbb{N}$, and $\mathbb{N}^*$ be the sets of all real numbers, all nonnegative, and positive integers, respectively. Suppose that we observe the output sequence $d_k \in \mathbb{R}$ ($k \in \mathbb{N}$) obeying the following linear measurement model:

$$d_k = \mathbf{u}_k^t \mathbf{h}_{\text{opt}} + n_k, \tag{1}$$

where $k \in \mathbb{N}$ denotes the time index, $N \in \mathbb{N}^*$ the tap length, $\mathbf{u}_k := [u_k, u_{k-1}, \ldots, u_{k-N+1}]^t \in \mathbb{R}^N$ an observed vector defined with the input sequence $u_k \in \mathbb{R}$, $\mathbf{h}_{\text{opt}}$ the unknown system we wish to estimate (e.g., echo impulse response), and $n_k \in \mathbb{R}$ the noise process ($(\cdot)^t$ stands for the transposition).

Moreover, we assume that the system is sparse, i.e., only a few coefficients of $\mathbf{h}_{\text{opt}}$ are significantly different from zero (active coefficients), and else are zero or near-zero (inactive coefficients) as shown in the left of Fig. 1. The objective is to approximate the unknown system $\mathbf{h}_{\text{opt}}$ (the support of the active coefficients is supposed to be unknown) by the adaptive filter $\mathbf{h}_k := [h_{1(k)}, h_{2(k)}, \ldots, h_{N(k)}]^t \in \mathbb{R}^N$ with the knowledge on $(\mathbf{u}_i, d_i)_{i=0}^k$ and an initial estimate $\mathbf{h}_0$ (see, the right of Fig. 1).

3. PROPOSED METHOD

3.1. Adaptively-Weighted Total Variation

Let $\mathbf{D}$ be a discrete gradient operator given by

$$\mathbf{D} : \mathbb{R}^N \to \mathbb{R}^{N-1} : h_{i(k)} \mapsto \begin{cases} h_{i+1(k)} - h_{i(k)}, & \text{if } i < N, \\ 0, & \text{if } i = N. \end{cases} \tag{2}$$

Then, we propose the *adaptively-weighted total variation* (AWTV) defined as follows:

$$\begin{aligned} \|\cdot\|_{TV}^{\mathbf{w}_k} &: \mathbb{R}^N \to [0, \infty) \\ &: \mathbf{h} \mapsto \|\mathbf{D}\mathbf{h}\|_1^{\mathbf{w}_k} = \sum_{i=1}^{N-1} w_{i(k)} |h_{i+1(k)} - h_{i(k)}|, \end{aligned} \tag{3}$$

where $\|\cdot\|_1^{\mathbf{w}_k}$ is AWℓ^1 introduced in [2], and $\mathbf{w}_k \in \mathbb{R}^{N-1}$ a weight vector containing $w_{i(k)} \in (0, \infty)$ ($i = 1, \ldots, N-1$). Each $w_{i(k)}$ is controlled to be a small value when the corresponding absolute difference $|h_{i+1(k)} - h_{i(k)}|$ is significantly large because such a difference represents the active coefficients of the unknown sparse system to be estimated, and hence it should be preserved. Indeed, each $w_{i(k)}$ is adaptively controlled as follows:

$$w_{i(k)} := \begin{cases} d\omega, & \text{if } |h_{i+1(k)} - h_{i(k)}| > t, \\ \omega, & \text{otherwise}, \end{cases} \tag{4}$$

where $\omega \in (0, \infty)$, $d \approx 0$, and $t > 0$ is the thresholding parameter.

To our best knowledge, there is no computationally-efficient technique for the calculation of the proximity operator of AWTV, which implies the difficulty of suppressing cost functions employing AWTV. On the other hand, the adaptive primal-dual splitting method to be presented in the next subsection can reduce its computation into the *time-varying soft thresholding* [2], resulting in a computationally efficient implementation.

3.2. Adaptive Primal-Dual Splitting Method

Let $\mathcal{H}, \mathcal{K}$ be real Hilbert spaces equipped with the standard inner products $\langle \cdot, \cdot \rangle_{\mathcal{H}}, \langle \cdot, \cdot \rangle_{\mathcal{K}}$ and their induced norms $\|\cdot\|_{\mathcal{H}}, \|\cdot\|_{\mathcal{K}}$, $\varphi_k, \psi_k \in \Gamma_0(\mathcal{H})$[1] ($k \in \mathbb{N}$), where φ_k is differentiable on $\mathcal{H}$ and its gradient $\nabla\varphi_k : \mathcal{H} \to \mathcal{H}$ is β_k-Lipschitzian[2] for some $\beta_k \in (0, \infty)$, $\vartheta_k \in \Gamma_0(\mathcal{K})$, and $L : \mathcal{H} \to \mathcal{K}$ a bounded linear operator. Consider the following time-varying cost function:

$$\Theta_k(\mathbf{x}) := \varphi_k(\mathbf{x}) + \psi_k(\mathbf{x}) + \vartheta_k(L\mathbf{x}). \tag{5}$$

Definition 3.1 (Adaptive Primal-Dual Splitting Method (APDS))**.** *For any $\mathbf{x}_0 \in \mathcal{H}$ and $\boldsymbol{\xi}_0 \in \mathcal{K}$, the adaptive primal-dual splitting method (APDS) for suppressing Θ_k is defined by*

$$\begin{cases} \hat{\mathbf{x}}_{k+1} := \operatorname{prox}_{\gamma\psi_k}[(I - \gamma\nabla\varphi_k)\mathbf{x}_k - \gamma L^* \boldsymbol{\xi}_k], \\ \hat{\boldsymbol{\xi}}_{k+1} := \operatorname{prox}_{\delta\vartheta_k^*}[\boldsymbol{\xi}_k + \delta L(2\mathbf{x}_{k+1} - \mathbf{x}_k)], \\ \mathbf{x}_{k+1} := (1 - \lambda_k)\mathbf{x}_k + \lambda_k \hat{\mathbf{x}}_{k+1}, \\ \boldsymbol{\xi}_{k+1} := (1 - \lambda_k)\boldsymbol{\xi}_k + \lambda_k \hat{\boldsymbol{\xi}}_{k+1}, \end{cases} \tag{6}$$

*where '*prox*' denotes the proximity operator*[3]*, ϑ_k^* the Fenchel-Rockafellar conjugate function*[4] *of ϑ_k, L^* the adjoint operator of L, $\gamma, \delta \in (0, \infty)$ satisfying that $\frac{1}{\gamma} - \delta\|L\|_{op}^2 > \frac{\beta_k}{2}$ ($\|\cdot\|_{op}$ stands for the operator norm), $\lambda_k \in [0, \frac{4\kappa - 1}{2\kappa}]$ such that $\sum_{k\in\mathbb{N}} \lambda_k (1 - \frac{2\kappa\lambda_k}{4\kappa - 1}) = \infty$, and $\kappa := \frac{1}{\beta_k}(\frac{1}{\gamma} - \delta\|L\|_{op}^2) > \frac{1}{2}$.*

Theorem 3.1 (Primal-Dual Monotone Approximation of APDS)**.** *Let $\Xi_k(\boldsymbol{\xi}) := (\varphi_k + \psi_k)^*(-L^*\boldsymbol{\xi}) + \vartheta_k^*(\boldsymbol{\xi})$, $\mathcal{Z} := \mathcal{H} \times \mathcal{K}$ be a real Hilbert space, where the inner product $\langle \cdot, \cdot \rangle_{\mathcal{Z}}$ and its induced norm $\|\cdot\|_{\mathcal{Z}}$ are defined by $\langle (\mathbf{x}, \boldsymbol{\xi}), (\mathbf{x}', \boldsymbol{\xi}') \rangle_{\mathcal{Z}} := \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathcal{H}} + \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathcal{K}}$ and $\|(\mathbf{x}, \boldsymbol{\xi})\|_{\mathcal{Z}} := \sqrt{\langle (\mathbf{x}, \boldsymbol{\xi}), (\mathbf{x}, \boldsymbol{\xi}) \rangle_{\mathcal{Z}}}$ for $(\mathbf{x}, \boldsymbol{\xi}), (\mathbf{x}', \boldsymbol{\xi}') \in \mathcal{Z}$. Furthermore, we define a bounded linear operator*

$$P : \mathcal{Z} \to \mathcal{Z} : \begin{pmatrix} \mathbf{x} \\ \boldsymbol{\xi} \end{pmatrix} \mapsto \begin{pmatrix} \frac{1}{\gamma} I & -L^* \\ -L & \frac{1}{\delta} I \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \boldsymbol{\xi} \end{pmatrix}, \tag{7}$$

which is self-adjoint, surjective, and $\forall (\mathbf{x}, \boldsymbol{\xi}) \in \mathcal{Z}$, $\exists a \in (0, \infty)$, $\langle (\mathbf{x}, \boldsymbol{\xi}), P(\mathbf{x}, \boldsymbol{\xi}) \rangle_{\mathcal{Z}} \geq a\|(\mathbf{x}, \boldsymbol{\xi})\|_{\mathcal{Z}}^2$. Then, we can define another real Hilbert space $\mathcal{Z}_P$ equipped with the inner product $\langle \cdot, \cdot \rangle_{\mathcal{Z}_P} := \langle \cdot, P\cdot \rangle_{\mathcal{Z}}$ and its induced norm $\|\cdot\|_{\mathcal{Z}_P}$.

Suppose that

$$\begin{aligned} &\bigcup_{\lambda > 0} \{\lambda \mathbf{x} \mid \mathbf{x} \in L\operatorname{dom}(\psi_k) - \operatorname{dom}(\vartheta_k)\} \\ &= \overline{\operatorname{span}}(L\operatorname{dom}(\psi_k) - \operatorname{dom}(\vartheta_k)), \end{aligned} \tag{8}$$

[1] A function $f : \mathcal{H} \to (-\infty, \infty]$ is called *proper lower semicontinuous convex* if $\operatorname{dom}(f) := \{\mathbf{x} \in \mathcal{H} \mid f(\mathbf{x}) < \infty\} \neq \emptyset$, $\operatorname{lev}_{\leq\alpha}(f) := \{\mathbf{x} \in \mathcal{H} \mid f(\mathbf{x}) \leq \alpha\}$ is closed for every $\alpha \in \mathbb{R}$, and $f(\lambda\mathbf{x} + (1 - \lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1 - \lambda) f(\mathbf{y})$ for every $\mathbf{x}, \mathbf{y} \in \mathcal{H}$ and $\lambda \in (0, 1)$, respectively. The set of all proper lower semicontinuous convex functions on $\mathcal{H}$ is denoted by $\Gamma_0(\mathcal{H})$.

[2] A mapping $T : \mathcal{H} \to \mathcal{H}$ is called *κ-Lipschitzian* if $\|T(\mathbf{x}) - T(\mathbf{y})\| \leq \kappa\|\mathbf{x} - \mathbf{y}\|$ for some $\kappa \in (0, \infty)$ and every $\mathbf{x}, \mathbf{y} \in \mathcal{H}$.

[3] For any $\gamma \in (0, \infty)$, the *proximity operator* of $f \in \Gamma_0(\mathcal{H})$ is given by

$$\operatorname{prox}_{\gamma f}(\mathbf{x}) := \arg\min_{\mathbf{y} \in \mathcal{H}} \left\{ f(\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{x} - \mathbf{y}\|^2 \right\}.$$

[4] The *Fenchel-Rockafellar conjugate function* of $f \in \Gamma_0(\mathcal{H})$ is defined by $f^*(\boldsymbol{\xi}) := \sup_{\mathbf{x} \in \mathcal{H}} \{\langle \mathbf{x}, \boldsymbol{\xi} \rangle - f(\mathbf{x})\}$. The proximity operator of f^* can be expressed as $\operatorname{prox}_{\gamma f^*}(\mathbf{x}) = \mathbf{x} - \gamma \operatorname{prox}_{\frac{1}{\gamma} f}(\frac{1}{\gamma}\mathbf{x})$.

and

$$(\mathbf{x}_k, \boldsymbol{\xi}_k) \notin \Omega_k := \left\{ (\mathbf{x}, \boldsymbol{\xi}) \in \mathcal{Z}_P \;\middle|\; \begin{smallmatrix} \Theta_k(\mathbf{x}) = \Theta_k^\star \\ \Xi_k(\boldsymbol{\xi}) = \Xi_k^\star \end{smallmatrix} \right\}, \tag{9}$$

where $\overline{span}(S)$ is the smallest closed subspace of $\mathcal{K}$ containing the set S, $\Theta_k^\star := inf_{\mathbf{x}\in\mathcal{H}} \Theta_k(\mathbf{x})$ and $\Xi_k^\star := inf_{\boldsymbol{\xi}\in\mathcal{K}} \Xi_k(\boldsymbol{\xi})$. Then, for any $(\mathbf{x}^{\star(k)}, \boldsymbol{\xi}^{\star(k)}) \in \Omega_k$, the sequence $\{(\mathbf{x}_k, \boldsymbol{\xi}_k)\}_{k\in\mathbb{N}}$ generated by the algorithm (6) satisfies

$$\begin{aligned} &\|(\mathbf{x}_{k+1}, \boldsymbol{\xi}_{k+1}) - (\mathbf{x}^{\star(k)}, \boldsymbol{\xi}^{\star(k)})\|_{\mathcal{Z}_P} \\ &< \|(\mathbf{x}_k, \boldsymbol{\xi}_k) - (\mathbf{x}^{\star(k)}, \boldsymbol{\xi}^{\star(k)})\|_{\mathcal{Z}_P}. \end{aligned} \tag{10}$$

The inequality (10) implies that $\{(\mathbf{x}_k, \boldsymbol{\xi}_k)\}_{k\geq 0}$ monotonically approaches the solution set Ω_k that is expected to include the unknown system to be estimated.

Remark 3.1 (Advantages of APDS compared with other proximal adaptive filtering methods)**.**

- APDS is able to suppress a time-varying cost function consisting of the sum of differentiable and multiple non-differentiable convex functions by using their gradient and proximity operators.
- APDS can deal with non-differentiable convex functions involving a linear operator, such as AWTV, without using operator inversion.

3.3. Example of Cost function Design

We design a time-varying cost function employing AWTV as follows:

$$\Theta_k^{TV}(\mathbf{h}) := \|\mathbf{h}\|_{TV}^{\mathbf{w}_k} + \iota_{S_k^{(\varepsilon_k)}}(\mathbf{h}), \tag{11}$$

where $\iota_{S_k^{(\varepsilon_k)}}$ is the *indicator function*[5] of the following nonempty closed convex set

$$S_k^{(\varepsilon_k)} := \{\mathbf{h} \in \mathbb{R}^N | \; |\mathbf{u}_k^t \mathbf{h} - d_k| \leq \varepsilon_k\}, \tag{12}$$

which is the so-called *hyper slab* [14] with a user-defined tolerance ε_k w.r.t. the additive noise $n_k \in \mathbb{R}$. The hyper slab $S_k^{(\varepsilon_k)}$ plays a role of a data-fidelity to the input-output pair $(\mathbf{u}_k, d_k)$ (also utilized in [8]). By letting

$$\varphi_k : \mathbb{R}^N \to \mathbb{R} : \mathbf{h} \mapsto 0, \tag{13}$$

$$\psi_k : \mathbb{R}^N \to [0, \infty] : \mathbf{h} \mapsto \iota_{S_k^{(\varepsilon_k)}}(\mathbf{h}), \tag{14}$$

$$\vartheta_k : \mathbb{R}^N \to [0, \infty] : \boldsymbol{\eta} \mapsto \|\boldsymbol{\eta}\|_1^{\mathbf{w}_k}, \tag{15}$$

$$L : \mathbb{R}^N \to \mathbb{R}^N : \mathbf{h} \mapsto \mathbf{D}\mathbf{h}, \tag{16}$$

in (5), the cost function (5) becomes equivalent to (11), so that APDS is applicable to (11), resulting in Algorithm 3.1.

Remark 3.2 (Note on The Implementation of Algorithm 3.1)**.**

- (Computation of $\mathbf{D}$ and $\mathbf{D}^t$) This can be implemented by the calculation of the difference between neighbouring filter coefficients, resulting in $\mathcal{O}(N)$ cost.
- (Computation of $\mathrm{prox}_{\frac{1}{\delta}\|\cdot\|_1^{\mathbf{w}_k}}$) The proximity operator of AWℓ^1 introduced in [2] is given by

$$\mathrm{prox}_{\frac{1}{\delta}\|\cdot\|_1^{\mathbf{w}_k}} : \mathbb{R}^N \to \mathbb{R}^N : x_i \mapsto \begin{cases} x_i - \frac{w_{i(k)}}{\delta} & \text{if } x_i > \frac{w_{i(k)}}{\delta}, \\ x_i & \text{if } -\frac{w_{i(k)}}{\delta} \leq x_i \leq \frac{w_{i(k)}}{\delta}, \\ x_i + \frac{w_{i(k)}}{\delta} & \text{if } x_i < -\frac{w_{i(k)}}{\delta}, \end{cases}$$

  which has $\mathcal{O}(N)$ cost.
- (Computation of $P_{S_k^{(\varepsilon_k)}}$) The projection onto the hyper slab $S_k^{(\varepsilon_k)}$, which has also $\mathcal{O}(N)$ cost, is given by

$$P_{S_k^{(\varepsilon_k)}} : \mathbb{R}^N \to \mathbb{R}^N : \mathbf{x} \mapsto \begin{cases} \mathbf{x}, & \mathbf{x} \in S_k^{(\varepsilon_k)}, \\ \mathbf{x} - \frac{(\mathbf{u}_k^t \mathbf{x} - d_k) - \mathrm{sgn}(\mathbf{u}_k^t \mathbf{x} - d_k)\varepsilon_k}{\|\mathbf{u}_k\|_2^2} \mathbf{u}_k, & \text{otherwise}, \end{cases}$$

  where 'sgn' denotes the signum function defined by

$$\mathrm{sgn} : \mathbb{R} \to \mathbb{R} : x \mapsto \begin{cases} \frac{x}{|x|}, & x \neq 0, \\ 0, & x = 0. \end{cases}$$

Hence, the total cost of the algorithm is $\mathcal{O}(N)$.

Algorithm 3.1 (APDS for (11))

Set $k = 0$, and choose $\mathbf{h}_0, \boldsymbol{\eta}_0, \mathbf{W}_0, \mu_0, \gamma_0, \delta_0$
while a stop criterion is not satisfied **do**
  $\mathbf{t}_k = \mathbf{h}_k - \gamma \mathbf{D}^t \boldsymbol{\eta}_k$
  $\hat{\mathbf{h}}_{k+1} = P_{S_k^{(\varepsilon_k)}}(\mathbf{t}_k)$
  $\boldsymbol{\tau}_k = \boldsymbol{\eta}_k + \delta \mathbf{D}(2\hat{\mathbf{h}}_{k+1} - \mathbf{h}_k)$
  $\hat{\boldsymbol{\eta}}_{k+1} = \boldsymbol{\tau}_k - \delta \mathrm{prox}_{\frac{1}{\delta}\|\cdot\|_1^{\mathbf{w}_k}}(\frac{1}{\delta}\boldsymbol{\tau}_k)$
  $\mathbf{h}_{k+1} = (1 - \lambda_k)\mathbf{h}_k + \lambda_k \hat{\mathbf{h}}_{k+1}$
  $\boldsymbol{\eta}_{k+1} = (1 - \lambda_k)\boldsymbol{\eta}_k + \lambda_k \hat{\boldsymbol{\eta}}_{k+1}$
  $k = k + 1$
end while

4. NUMERICAL EXPERIMENT

We examined the performance of APDS with AWTV in the context of a simple network echo cancellation problem for white noise input. We used the sparse echo impulse response $\mathbf{h}_{\mathrm{opt}}$ of length $N = 512$ with sampling rate 8 kHz initialized according to the model 1 of ITU-T G.168 [15], shown in the left of Fig. 1. The input signal $\mathbf{u}_k$ was generated according $\mathcal{N}(0, 1)$. The noise n_k was set to zero mean white Gaussian with signal-to-noise ratio (SNR)=15dB, where SNR:$= 10\log_{10}(E[(\mathbf{u}_k^t \mathbf{h}_{\mathrm{opt}})^2]/E[n_k^2])$.

Methods for comparison are listed in Remark 4.1. Their step-sizes were chosen in such a way that their convergence speed are the same (all the following methods have $\mathcal{O}(N)$ cost).

Remark 4.1 (Methods for Comparison)**.**

- **'NLMS'**: It stands for the Normalized Least Mean Square (NLMS) [16] with the step-size 1. NLMS is interpreted as an algorithm which iteratively performs the projection onto $S_k^{(0)}$ (see (12)).

[5] For a given nonempty closed convex set C in a real Hilbert space $\mathcal{H}$, its *indicator function* is defined as

$$\iota_C(\mathbf{x}) := \begin{cases} 0, & \text{if } \mathbf{x} \in C, \\ \infty, & \text{otherwise.} \end{cases}$$

The proximity operator of ι_C for any $\gamma \in (0, \infty)$ coincides with the metric projection onto C, i.e., $\mathrm{prox}_{\gamma\iota_C}(\mathbf{x}) = P_C(\mathbf{x}) := \underset{\mathbf{y}\in C}{\mathrm{argmin}} \|\mathbf{y} - \mathbf{x}\|$.

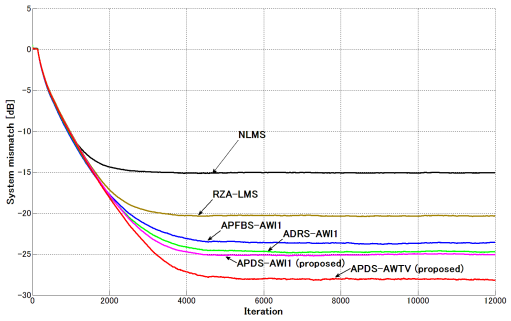

Fig. 2. Comparison of the methods in system mismatch.

- **'RZA-LMS'**: It stands for the Reweighted Zero-Attracting (RZA) LMS [1][6] with the step-size 0.7. The parameters were set as $(\delta, \lambda, c_{RZA}) = (1, 4.0 \times 10^{-4}, 1.0 \times 10^{5})$.
- **'APFBS-AWl1'**: It stands for APFBS employing AWℓ_1 [2] with the step-size 0.9, where the cost function is the sum of AWℓ_1 and the square of the distance function w.r.t. the set $S_k^{(\varepsilon_k)}$. The parameters were set as $(\omega, d, t, \varepsilon_k, \gamma) = (1, 1.0 \times 10^{-6}, 5.0 \times 10^{-4}, 4.2 \times 10^{-2}, 1)$.
- **'ADRS-AWl1'**: It stands for ADRS employing AWℓ_1 [8] with the step-size 1.7. The cost function is given by

$$\Theta_k^{\ell_1}(\mathbf{h}) := \|\mathbf{h}\|_1^{\mathbf{w}_k} + \iota_{S_k^{(\varepsilon_k)}}(\mathbf{h}), \tag{18}$$

  where the weight $\mathbf{w}_k$ is controlled by the technique in [2]. The parameters were set as $(\omega, d, t, \varepsilon_k, \gamma) = (1, 1.0 \times 10^{-6}, 5.0 \times 10^{-4}, 4.2 \times 10^{-2}, 1)$.
- **'APDS-AWl1'**: It stands for APDS employing AWℓ_1 with the step-size 0.8. The cost function is given by (18). The parameters were set as $(\omega, d, t, \varepsilon_k, \gamma, \delta) = (8, 1.0 \times 10^{-6}, 8.0 \times 10^{-4}, 4.2 \times 10^{-2}, 0.15, 0.15)$. This is for comparison of the efficacy of AWℓ_1 and AWTV.
- **'APDS-AWTV'**: It stands for APDS employing AWTV with the step-size 0.8. The cost function is given by (11). The parameters were set as $(\omega, d, t, \varepsilon_k, \gamma, \delta) = (8, 1.0 \times 10^{-6}, 8.0 \times 10^{-4}, 4.2 \times 10^{-2}, 0.15, 0.15)$.

Figure 2 depicts a comparison of the methods in the sense of system-mismatch $10 \log_{10} \frac{\|\mathbf{h}_{\text{opt}} - \mathbf{h}_k\|_2^2}{\|\mathbf{h}_{\text{opt}}\|_2^2}$ averaged over 100 runs. 'APDS-TV' (proposed) achieved the best stead-state behavior. This result indicates that AWTV is much effective for estimating sparse systems compared with AWℓ^1. It suggests that the suppression of AWTV brings efficient smoothing to the inactive coefficients, which means that it more quickly pushes them down to zero than the suppression of AWℓ^1, while keeping the active coefficients. At the same time, APDS itself seems to be an efficient adaptive filtering method from the comparison of ADRS-AWl1' and 'APDS-AWl1', where APDS indicates a better performance even they use the same cost function. This may be because of the monotone approximation property of APDS, which ADRS does not have.

One may think that the system used in this experiment is group sparse, so that group ℓ^1 norms [17, 18, 19] can be also considered as a suitable choice for sparsity-inducing term. An advantage of AWTV compared to them is that it does not require information on the support of the active coefficients of the unknown system.

We should consider the case that the system to be estimated is sparse but highly non-smooth, i.e., the positions of the active coefficients are completely random. In such a case, AWTV may not be as effective as AWℓ^1 because the value of AWTV is approximately twice as large as that of AWℓ^1.

Although we fixed the parameters of APDS (and the other methods) in each time step in this experiment, it is possible to control them in some online manner, for example, the parameter ω can be updated in such a way that it is inversely proportional to the value of AWTV in the last time step, which enables us to avoid over-smoothing in the case that the system to be estimated is highly non-smooth.

5. CONCLUDING REMARKS

We have proposed the adaptively-weighted total variation (AWTV) and the adaptive primal-dual splitting algorithm (APDS), for sparse system identification. AWTV was designed to exploit the smoothness of sparse systems in online manner. APDS is a computationally-efficient adaptive algorithm for dealing with time-varying cost functions which consist of the sum of differentiable and multiple non-differentiable convex functions with the composition of linear operators. Its primal-dual monotone approximation property guaranteed that the sequence of APDS approaches the solution set of the current cost function in each time-step. We have also presented a useful example of the cost function of APDS employing AWTV on sparse system identification. In the following, we give a brief discussion on how our main contributions (AWTV and APDS) are related to prior work.

As mentioned in Section 1, AWTV is an adaptive extension of the total variation (TV) [11] that has been a popular tool in signal and image processing fields. Advanced work on TV is found, for example, in [20, 21, 22, 23, 24]. However, it has not been developed for sparse system identification, and in this sense, our proposed AWTV broadens the applicability of TV.

APDS is categorized as a proximal adaptive filtering method which can efficiently suppress non-differentiable convex cost functions by using the notion of proximity operator. Such a method was first proposed in [2] known as APFBS, and it has been extended in [6, 7, 9, 10]. ADRS [8] is also one of them and the only method, except APDS, being able to handle cost functions employing multiple non-differentiable convex functions. APDS is regarded as an advanced method compared with APFBS and APDS in the sense described in Remark 3.1. APDS offers wide range of further applications considering sparsity, such as kernel adaptive filtering [25] and distributed learning [26, 27]. At the same time, APDS can impose various types of convex constraints, including the weighted ℓ_1 ball [28], the nonnegative constraint [29], and other useful convex sets [30], on the cost function via their indicator functions. Of course, it can also handle a variety of other convex priors, such as the $\ell^{1,2}$ and $\ell^{1,\infty}$ norms for promoting group sparsity [17, 18, 19] and the Huber loss function [31] for being robust to impulsive noise [32, 33, 26, 10]. Finally, we remark again that APDS is a time-varying extension of the primal-dual splitting algorithm [12].

[6] RZA-LMS is described by the following equation:

$$\mathbf{h}_{k+1} := \mathbf{h}_k + \mu \frac{\mathbf{u}_k^t \mathbf{h}_k - d_k}{\|\mathbf{u}_k\|_2^2 + \delta} \mathbf{u}_k - \lambda \sum_{i=1}^{N} \frac{\text{sgn}((\mathbf{h}_k)_i)}{1 + c_{RZA}|(\mathbf{h}_k)_i|} \mathbf{e}_i, \tag{17}$$

where $\{\mathbf{e}_i\}_{i=1}^N$ is the standard orthonormal basis of $\mathbb{R}^N$, i.e., $\mathbf{e}_i := [0, \ldots, 0, 1, 0, \ldots, 0]^t$ with the value 1 assigned to its i-th position, $\mu \in (0, \infty)$ the step-size, $\delta \in [0, \infty)$ the parameter for numerical stability, $\lambda \in (0, \infty)$ the sparsity parameter, and $c_{RZA} \in (0, \infty)$ is a constant.

6. REFERENCES

[1] Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. IEEE ICASSP*, 2009.

[2] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. IEEE ICASSP*, 2010.

[3] J. Jin, Y. Gu, and S. Mei, "A stochastic gradient approach on compressive sensing signal reconstruction based on adaptive filtering framework," *IEEE J. Sel. Topics. Signal Process.*, vol. 4, no. 2, pp. 409–420, 2010.

[4] G. Mileounis, B. Babadi, N. Kalouptsidis, and V. Tarokh, "An adaptive greedy algorithm with application to nonlinear communications," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 2998–3007, 2010.

[5] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The sparse RLS algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 4013–4025, 2010.

[6] M. Yamagishi, M. Yukawa, and I. Yamada, "Sparse system identification by exponentially weighted adaptive parallel projection and generalized soft-thresholding," in *Proc. APSIPA ASC*, 2010.

[7] 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.

[8] I. Yamada, S. Gandy, and M. Yamagishi, "Sparsity-aware adaptive filtering based on a Douglas-Rachford splitting," in *Proc. EUSIPCO*, 2011.

[9] M. Yukawa, Y. Tawara, M. Yamagishi, and I. Yamada, "Sparsity-aware adaptive filters based on Lp-norm inspired soft-thresholding technique," in *Proc. IEEE ISCAS*, 2012.

[10] T. Yamamoto, M. Yamagishi, and I. Yamada, "Adaptive proximal forward-backward splitting for sparse system identification under impulsive noise," in *Proc. EUSIPCO*, 2012.

[11] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1-4, pp. 259–268, 1992.

[12] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optimization Theory and Applications*, 2012, DOI 10.1007/s10957-012-0245-9.

[13] S. Ono and I. Yamada, "A convex regularizer for reducing color artifact in color image recovery," in *Proc. of CVPR*, 2013, (accepted).

[14] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y. F. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Process. Lett.*, vol. 5, no. 5, pp. 111–114, 1998.

[15] Digital Network Echo Cancellers, *ITU-T Rec. G.168.*, 2007.

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

[17] M. Yuan andd Y. Lin, "Model selection and estimation in regression with grouped variables," *J. R. Statist. Soc. B*, vol. 70, no. 1, pp. 49–67, 2006.

[18] H. Wang and C. Leng, "A note on adaptive group lasso," *Computational Statistics & Data Analysis*, vol. 52, no. 12, pp. 5277–5286, 2008.

[19] Y. Chen and A. O. Hero, "Recursive $\ell_{1,\infty}$ group lasso," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3978–3987, 2012.

[20] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imaging Sci.*, vol. 3, no. 3, pp. 92–526, 2010.

[21] J. M. Fadili and G. Peyré, "Total variation projection with first order schemes," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 657–669, 2011.

[22] F. I. Karahanoğlu, İ. Bayram, and D. V. D. Ville, "A signal processing approach to generalized 1-D total variation," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5265, 2011.

[23] D. Q. Chen and L. Z. Cheng, "Spatially adapted total variation model to remove multiplicative noise," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1650–1662, 2012.

[24] Y. Hu and M. Jacob, "Higher degree total variation (HDTV) regularization for image recovery," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2559–2571, 2012.

[25] M. Yukawa, "Multikernel adaptive filtering," *IEEE Trans. Signal Process.*, vol. 60, no. p, pp. 4672–4682, 2012.

[26] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4692–4707, 2011.

[27] P. Di Lorenzo and A. H. Sayed, "Sparse distributed learning based on diffusion adaptation," *IEEE Trans. Signal Process. (to appear: available at arXiv:1206.3099)*, 2013.

[28] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online sparse system identification and signal reconstruction using projections onto weighted l1 balls," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 936–952, 2011.

[29] J. Chen, C. Richard, J. C. M. Bermudez, and P. Honeine, "Nonnegative least-mean-square algorithm," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5225–5235, 2011.

[30] 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. Magazine*, vol. 28, no. 1, pp. 97–123, 2011.

[31] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Statist.*, vol. 35, pp. 73–101, 1964.

[32] P. Petrus, "Robust Huber adaptive filter," *IEEE Trans. Signal Process.*, vol. 47, no. 1, pp. 1129–1133, 1999.

[33] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A new robust variable step-size NLMS algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1878–1893, 2008.