SPARSITY-AWARE ADAPTIVE FILTERING BASED ON A DOUGLAS-RACHFORD SPLITTING

Isao Yamada, Silvia Gandy, and Masao Yamagishi

Department of Communications and Integrated Systems,
Tokyo Institute of Technology, 2-12-1-S3-60 Ookayama, Meguro-ku, Tokyo 152-8550, Japan
email: {isao, gandy, myamagi}@sp.ss.titech.ac.jp

ABSTRACT

In this paper, we propose a novel online scheme for the sparse adaptive filtering problem. It is based on a formulation of the adaptive filtering problem as a minimization of the sum of (possibly *nonsmooth*) convex functions. Our proposed scheme is a time-varying extension of the so-called *Douglas-Rachford splitting method*. It covers many existing adaptive filtering algorithms as special cases. We show several examples of special choices of the cost functions that reproduce those existing algorithms. Our scheme achieves a monotone decrease of an upper bound of the distance to the solution set of the minimization under certain conditions. We applied a simple algorithm that falls under our scheme to a sparse echo cancellation problem where it shows excellent convergence performance.

1. INTRODUCTION

Recently, there has been an increased interest in developing adaptive filtering algorithms which exploit the *sparsity* of the unknown system in its estimation. Many adaptive filtering algorithms minimize a time-varying cost function, in the sense of keeping the value as low as possible, to obtain a replica of the unknown system. The sparsity prior is utilized by incorporating a sparsity-inducing term (usually a *nonsmooth* convex function) into the cost function, for instance adding a term that includes the (weighted) ℓ_1 norm (see for example [3, 17, 23, 28]). One example for such an algorithm is the *adaptive proximal forward-backward splitting* (APFBS) scheme [17, 28], which is a time-varying extension of the *proximal forward-backward splitting* [6, 20]. The APFBS scheme attempts to minimize a time-varying cost function which is a sum of one smooth and one nonsmooth convex function. It covers many conventional standard/proportionate-type algorithms, for example, NLMS [18]/APA [12, 19] and PNLMS [7]/PAPA [1, 10]. An acceleration of APFBS was proposed [29].

Still, the cost functions in most adaptive filtering algorithms are restricted to the following case. The cost function has to be the sum of a smooth convex function and a nonsmooth convex function. Hence it is of great interest to extend the class of cost functions applicable in adaptive filtering algorithms, thereby giving way to novel adaptive filtering algorithms.

In this paper, we propose an adaptive filtering scheme which can utilize the sum of multiple nonsmooth convex functions. The general scheme is based on the application of the so-called *Douglas-Rachford splitting* algorithm [4, 27] which minimizes the sum of two (possibly nonsmooth) convex functions by the iterative use of the *proximity operator* [6, 16] of each convex function. We extended the Douglas-Rachford splitting algorithm to the setting of a time-varying cost function.

The adaptive filter in our scheme is defined as an application of the proximity operator to an auxiliary sequence. This sequence has the nice property that an upper bound of its distance to the inverse image (w.r.t. the proximal map) of the minimizer of the cost function decreases monotonically in each time-step.

This scheme reproduces well-known adaptive filtering algorithms, for example, NLMS [18]/APA [12, 19], PNLMS [7]/PAPA [1, 10] and the APFBS scheme [17, 28] by setting the time-varying cost function accordingly. Sparsity-aware adaptive filtering algorithms within our scheme are obtained for instance by incorporating a time-varying weighted ℓ_1 norm.

This work was partially supported by SCAT (Support Center for Advanced Telecommunications) and JSPS Grants-in-Aid (09J09539).

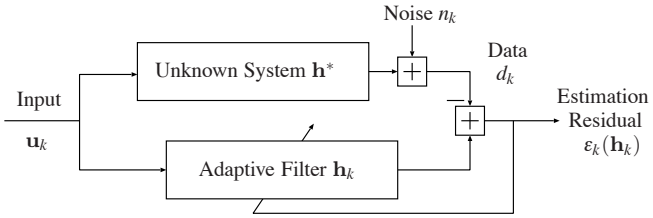

Figure 1: Adaptive filtering scheme.

We can further generalize our scheme to the case where the cost function is the sum of more than two convex functions by making use of a product space formulation [5, 8, 9, 21]. The result is an online scheme for the minimization of the sum of multiple time-varying convex functions. As an example of this scheme, we propose a sparsity-aware adaptive learning in transform domain.

As a numerical example, we present a sparse echo cancellation setting. In this example, a simple algorithm that falls under our scheme shows excellent convergence performance.

2. SPARSITY-AWARE ADAPTIVE FILTERING PROBLEM

Let $\mathbb{R}$ and $\mathbb{N}$ denote the sets of all real numbers and nonnegative integers, respectively. Denote the set $\mathbb{N}\setminus\{0\}$ by $\mathbb{N}^*$ and transposition of a matrix or a vector by $(\cdot)^T$. Suppose that we observe the output sequence $d_k \in \mathbb{R}$ ($k \in \mathbb{N}$) that obeys the following model (see Fig. 1):

$$d_k = \mathbf{u}_k^T \mathbf{h}^* + n_k,$$

where $k \in \mathbb{N}$ denotes the time index, $N \in \mathbb{N}^*$ the tap length, $\mathbf{u}_k := [u_k, u_{k-1}, \ldots, u_{k-N+1}]^T \in \mathbb{R}^N$ a known vector defined with the input sequence $u_k \in \mathbb{R}$ ($k \in \mathbb{N}$), $\mathbf{h}^* \in \mathbb{R}^N$ the unknown system to be estimated (e.g., echo impulse response), and $n_k \in \mathbb{R}$ the noise process. Throughout this paper we consider $\mathbf{h}^* \in \mathbb{R}^N$ to be *sparse*, i.e., few coefficients are significantly different from zero (active coefficients) and many coefficients are zero or near-zero (inactive coefficients).

For a finite number of measurements $r \in \mathbb{N}^*$ (usually $r \ll N$), the sequence $(d_k)_{k\in\mathbb{N}} \subset \mathbb{R}$ can be written compactly in the following form:

$$\mathbf{d}_k = U_k^T \mathbf{h}^* + \mathbf{n}_k,$$

where $\mathbf{d}_k := [d_k, d_{k-1}, \ldots, d_{k-r+1}]^T \in \mathbb{R}^r$, $U_k := [\mathbf{u}_k, \mathbf{u}_{k-1}, \ldots, \mathbf{u}_{k-r+1}] \in \mathbb{R}^{N\times r}$, and the noise vector $\mathbf{n}_k := [n_k, n_{k-1}, \ldots, n_{k-r+1}]^T \in \mathbb{R}^r$ for all $k \in \mathbb{N}$. In addition, we define the estimation residual functions $\varepsilon_k : \mathbb{R}^N \to \mathbb{R}^r$ for $k \in \mathbb{N}$ by

$$\varepsilon_k(\mathbf{h}) := U_k^T \mathbf{h} - \mathbf{d}_k,\ \mathbf{h} \in \mathbb{R}^N. \tag{1}$$

A major goal of the adaptive filtering problem is to approximate the unknown system $\mathbf{h}^*$ 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 $(\mathbf{u}_i, \mathbf{d}_i)_{i=0}^k$ and an initial estimate $\mathbf{h}_0 \in \mathbb{R}^N$.

3. TIME-VARYING EXTENSION OF DOUGLAS-RACHFORD SPLITTING METHOD

For every $k \in \mathbb{N}$, let $Q_k \in \mathbb{R}^{N\times N}$ be a symmetric and positive definite matrix which is used to define the inner product $\langle \mathbf{x}, \mathbf{y}\rangle_{Q_k} := \mathbf{x}^T Q_k \mathbf{y}$

and its induced norm $\|\mathbf{x}\|_{Q_k} := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_{Q_k}}$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$. We consider the situation where the unknown system $\mathbf{h}^* \in \mathbb{R}^N$ can be expected to lie in the neighborhood of $\Omega_k := \operatorname*{arg\,min}_{\mathbf{h} \in \mathbb{R}^N} \Theta_k(\mathbf{h}) \neq \varnothing$, where $\Theta_k \colon \mathbb{R}^N \to (-\infty, \infty]$, $k \in \mathbb{N}$ are time-varying cost functions. Therefore our goal is to track the set Ω_k with the adaptive filter $\mathbf{h}_k$. Most adaptive filtering algorithms implicitly utilize this simple idea.

3.1 Proposed scheme for the sum of two convex functions

Suppose that Θ_k can be decomposed as the sum of two functions:

$$\Theta_k(\mathbf{h}) := \varphi_k(\mathbf{h}) + \psi_k(\mathbf{h}), \tag{2}$$

where $\varphi_k \colon \mathbb{R}^N \to (-\infty, \infty]$ and $\psi_k \colon \mathbb{R}^N \to (-\infty, \infty]$ are proper lower semicontinuous convex functions (see for example [13]). We additionally suppose that the proximity operators[1] $\operatorname{prox}^{(Q_k)}_{\gamma_k \varphi_k}$ and $\operatorname{prox}^{(Q_k)}_{\gamma_k \psi_k}$ of φ_k and ψ_k can be computed efficiently.

In order to minimize the time-varying function Θ_k in an online way, we propose a time-varying extension of the *Douglas-Rachford splitting method*.

Algorithm 1 (Adaptive Douglas-Rachford splitting algorithm) *For an arbitrary initial vector $\mathbf{g}_0 \in \mathbb{R}^N$, generate a sequence $\mathbf{h}_k \in \mathbb{R}^N$ $(k \in \mathbb{N})$ by*

$$\mathbf{h}_k := \operatorname{prox}^{(Q_k)}_{\gamma_k \psi_k}(\mathbf{g}_k) \tag{3}$$

with

$$\mathbf{g}_{k+1} := \mathbf{g}_k + t_k \left\{ \operatorname{prox}^{(Q_k)}_{\gamma_k \varphi_k}(2\mathbf{h}_k - \mathbf{g}_k) - \mathbf{h}_k \right\}, \tag{4}$$

where $\gamma_k \in (0, \infty)$ and $t_k \in (0, 2)$ $(k \in \mathbb{N})$.

The next proposition is obtained by Proposition 18 in [4] and the firm nonexpansivity of the standard proximity operator.

Proposition 1 (Properties of Algorithm 1) *Suppose that the functions φ_k and ψ_k satisfy the* qualification condition[2] *for every $k \in \mathbb{N}$.*

Then the sequences $(\mathbf{h}_k)_{k \in \mathbb{N}}$ and $(\mathbf{g}_k)_{k \in \mathbb{N}}$ generated by Algorithm 1 satisfy the following

(i)

$$\begin{cases} \left\| \mathbf{h}_{k+1} - \operatorname{prox}^{(Q_{k+1})}_{\gamma_{k+1} \psi_{k+1}}(\mathbf{g}^*_{k+1}) \right\|_{Q_{k+1}} \leq \|\mathbf{g}_{k+1} - \mathbf{g}^*_{k+1}\|_{Q_{k+1}} \\ \|\mathbf{g}_{k+1} - \mathbf{g}^*_{k+1}\|_{Q_{k+1}} \leq \sqrt{\dfrac{\lambda_{\max}(Q_{k+1})}{\lambda_{\min}(Q_k)}} \|\mathbf{g}_{k+1} - \mathbf{g}^*_{k+1}\|_{Q_k} \\ \|\mathbf{g}_{k+1} - \mathbf{g}^*_k\|_{Q_k} < \|\mathbf{g}_k - \mathbf{g}^*_k\|_{Q_k} \end{cases}$$

for all $\mathbf{g}^*_{k+i} \in \left(\operatorname{prox}^{(Q_{k+i})}_{\gamma_{k+i} \psi_{k+i}} \right)^{-1}(\Omega_{k+i})$ $(i = 0, 1)$, where $\lambda_{\max}$ and $\lambda_{\min}$ denote respectively the maximum and the minimum eigenvalues of a matrix.

(ii) Suppose there exists a $N_0 \in \mathbb{N}$ such that $\Omega := \cap_{i \geq N_0} \Omega_i \neq \varnothing$, $Q_i = Q$, $\psi_i = \psi$, and $\gamma_i = \gamma$ for all $i \geq N_0$. Then we have

$$\left\| \mathbf{h}_{k+1} - \operatorname{prox}^{(Q)}_{\gamma \psi}(\mathbf{g}^*) \right\|_Q \leq \|\mathbf{g}_{k+1} - \mathbf{g}^*\|_Q < \|\mathbf{g}_k - \mathbf{g}^*\|_Q$$

for all $k \geq N_0$ and all $\mathbf{g}^* \in \left(\operatorname{prox}^{(Q)}_{\gamma \psi} \right)^{-1}(\Omega)$.

(iii) (Convergence of the Douglas-Rachford splitting method *[4]) Suppose that $Q_k = Q$, $\varphi_k = \varphi$, $\psi_k = \psi$ (i.e., $\Omega_k = \Omega$) and $\gamma_k = \gamma$ for all $k \in \mathbb{N}$. Then by using $(t_k)_{k \in \mathbb{N}}$ satisfying $\sum_{k \in \mathbb{N}} t_k(2 - t_k) = \infty$, we have*

$$\left\| \mathbf{h}_k - \operatorname{prox}^{(Q)}_{\gamma \psi}(\mathbf{g}^*) \right\|_Q \leq \|\mathbf{g}_k - \mathbf{g}^*\|_Q \overset{k \to \infty}{\longrightarrow} 0$$

for some $\mathbf{g}^* \in \left(\operatorname{prox}^{(Q)}_{\gamma \psi} \right)^{-1}(\Omega)$.

Note that Proposition 1(ii) implies a monotone decrease of a sequence of upper bounds $(\|\mathbf{g}_k - \mathbf{g}^*\|_Q)_{k \in \mathbb{N}}$ of the distance[3] $d_Q(\mathbf{h}_k, \Omega)$ without assuming $\varphi_i = \varphi$ for any $i \geq N_0$. This property is useful for adaptive filtering applications.

In the following section we will present some useful examples of (φ_k, ψ_k) for adaptive filtering applications.

3.2 Useful choices of φ_k and ψ_k

Example 1 (Indicator function) We propose to use the indicator function indicator function

$$\iota_{S_k} \colon \mathbb{R}^N \ni \mathbf{h} \mapsto \begin{cases} 0 & \text{if } \mathbf{h} \in S_k \\ \infty & \text{otherwise.} \end{cases} \tag{5}$$

as φ_k or ψ_k in the objective function Θ_k. Here, $(S_k)_{k \in \mathbb{N}}$ is a sequence of closed convex sets. These sets S_k will represent the sets of candidate solutions at time k in the adaptive filtering problem. It is easy to show, that the proximity operator of ι_{S_k} is identical to the metric projection onto the closed convex set S_k, i.e.,

$$\operatorname{prox}^{(Q_k)}_{\gamma_k \iota_{S_k}}(\mathbf{g}) = \operatorname*{arg\,min}_{\mathbf{x} \in S_k} \|\mathbf{x} - \mathbf{g}\|_{Q_k} =: P^{(Q_k)}_{S_k}(\mathbf{g}).$$

Algorithm 1 with the selection of ψ_k as in (5) achieves $\mathbf{h}_k \in S_k$ for any selection of φ_k and for every $k \in \mathbb{N}$. In other words, Algorithm 1 can utilize S_k as a hard constraint as well as a nonsmooth term φ_k which represents a priori knowledge on $\mathbf{h}^*$.

Algorithm 1 with the selection of φ_k as in (5) covers many existing algorithms. This becomes apparent when also specifying S_k and ψ_k.

Example 2 (Useful sets S_k) Define the sets S_k in (5) as

$$S_k := \operatorname*{arg\,min}_{\mathbf{h} \in \mathbb{R}^N} \|\varepsilon_k(\mathbf{h})\|_2, \tag{6}$$

where we use $\|\cdot\|_2$ for the standard Euclidean norm on $\mathbb{R}^r$. Moreover, we assume that U_k has full column rank and Q_k is a diagonal matrix, i.e., $Q_k := \operatorname{diag}\{q_1^{(k)}, q_2^{(k)}, \ldots, q_N^{(k)}\}$ (Note that many of the variable metric projection type methods, such as the proportionate NLMS/APA [1, 7], utilize diagonal matrices (see [30])). Then, the update (4) in Algorithm 1 with $\varphi_k := \iota_{S_k}$ in (5) is reduced to

$$\mathbf{g}_{k+1} := \mathbf{g}_k - t_k Q_k^{-1} U_k \Gamma_k^{-1} \varepsilon_k(2\mathbf{h}_k - \mathbf{g}_k) + t_k(\mathbf{h}_k - \mathbf{g}_k), \tag{7}$$

where $\Gamma_k := U_k^T Q_k^{-1} U_k + \delta I$. The regularization parameter $\delta \geq 0$ is introduced for numerical stability. Algorithm 1 with (7) significantly extends the standard NLMS/APA [12, 19] and the proportionate NLMS/APA [1, 7]. APA, for instance, is reproduced by setting $Q_k := I$ and $\psi_k := 0$, where I is the identity matrix.

[1] The proximity operator $\operatorname{prox}^{(Q_k)}_{\gamma_k \varphi}$ of a proper lower-semicontinuous function φ of index $\gamma_k > 0$ and norm $\|\cdot\|_{Q_k}$ is defined as

$$\operatorname{prox}^{(Q_k)}_{\gamma_k \varphi}(\mathbf{g}) := \operatorname*{arg\,min}_{\mathbf{x} \in \mathbb{R}^N} \left(\varphi(\mathbf{x}) + \frac{1}{2\gamma_k} \|\mathbf{x} - \mathbf{g}\|^2_{Q_k} \right), \forall \mathbf{g} \in \mathbb{R}^N.$$

[2] Qualification condition [4]: The set

$$\operatorname{cone}(\operatorname{dom}(\varphi_k) - \operatorname{dom}(\psi_k)) := \bigcup_{\lambda > 0} \{\lambda \mathbf{x} \mid \mathbf{x} \in \operatorname{dom}(\varphi_k) - \operatorname{dom}(\psi_k)\}$$

is a subspace of $\mathbb{R}^N$, where

$$\operatorname{dom}(\varphi_k) - \operatorname{dom}(\psi_k) := \{\mathbf{x}_1 - \mathbf{x}_2 \in \mathbb{R}^N \mid \forall (\mathbf{x}_1, \mathbf{x}_2) \in \operatorname{dom}(\varphi_k) \times \operatorname{dom}(\psi_k)\}.$$

[3] The distance between an arbitrary point $\mathbf{x} \in \mathbb{R}^N$ and a closed convex set $C \subset \mathbb{R}^N$ is defined by $d_{Q_k}(\mathbf{x}, C) := \min_{\mathbf{y} \in C} \|\mathbf{x} - \mathbf{y}\|_{Q_k}$.

Another selection of S_k is a *hyperslab*

$$S_k := \{\mathbf{h} \in \mathbb{R}^N \mid |\mathbf{u}_k^T \mathbf{h} - d_k| \le \zeta_k\} \tag{8}$$

for some user-defined tolerance $\zeta_k > 0$. This set S_k has a closed-form expression of $P_{S_k}^{(Q_k)}$:

$$P_{S_k}^{(Q_k)}(\mathbf{h}) = \mathbf{h} - \begin{cases} 0 & \mathbf{h} \in S_k, \\ \frac{\varepsilon_k(\mathbf{h}) - \mathrm{sgn}(\varepsilon_k(\mathbf{h}))\zeta_k}{\|Q_k^{-1}\mathbf{u}_k\|_{Q_k}^2} Q_k^{-1}\mathbf{u}_k & \text{otherwise} \end{cases}$$

with $\varepsilon_k(\mathbf{h})$ in (1) for $r = 1$, where $\mathrm{sgn}(\cdot)$ is the signum function defined by $\mathrm{sgn}(x) := x/|x|$ if $x \neq 0$, $\mathrm{sgn}(x) := 0$ otherwise, for all $x \in \mathbb{R}$,

Example 3 (The adaptive proximal forward-backward splitting scheme) Let $g_k \colon \mathbb{R}^N \to (-\infty, \infty]$ be a lower semicontinuous convex function and $f_k \colon \mathbb{R}^N \to \mathbb{R}$ a smooth convex function whose gradient $\nabla_{(Q_k)} f_k$ is Lipschitz continuous with Lipschitz constant L_k, i.e.,

$$\|\nabla_{(Q_k)} f_k(\mathbf{x}) - \nabla_{(Q_k)} f_k(\mathbf{y})\|_{Q_k} \le L_k \|\mathbf{x} - \mathbf{y}\|_{Q_k}, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^N.$$

Define φ_k of the objective function Θ_k by a upper bound of $f_k + g_k$:

$$\varphi_k(\mathbf{h}) := g_k(\mathbf{h}) + f_k(\mathbf{h}_k) + \langle \nabla_{(Q_k)} f_k(\mathbf{h}_k), \mathbf{h} - \mathbf{h}_k \rangle_{Q_k} + \frac{L_k}{2} \|\mathbf{h} - \mathbf{h}_k\|_{Q_k}^2. \tag{9}$$

It is easy to show, that the proximity operator of φ_k in (9) at $\mathbf{h}_k$ is identical to the so-called *proximal gradient operator* (for example [6]) at $\mathbf{h}_k$, i.e., $\mathrm{prox}_{\gamma_k \varphi_k}^{(Q_k)}(\mathbf{h}_k) = \mathrm{prox}_{\beta_k g_k}^{(Q_k)}\left(\mathbf{h}_k - \beta_k \nabla_{(Q_k)} f_k(\mathbf{h}_k)\right)$ with $\beta_k = (L_k + \gamma_k^{-1})^{-1}$. Hence the *adaptive proximal forward-backward splitting scheme* [17, 28] is reproduced as an example of Algorithm 1 by setting $\psi_k := 0$ and $t_k := 1$, i.e.,

$$\mathbf{h}_{k+1} := \mathrm{prox}_{\beta_k g_k}^{(Q_k)}\left(\mathbf{h}_k - \beta_k \nabla_{(Q_k)} f_k(\mathbf{h}_k)\right). \tag{10}$$

The available stepsize range of β_k in (10) is reduced compared with the stepsize range $(0, 2L_k^{-1})$ of the original adaptive proximal forward-backward splitting scheme.

In particular, if we choose

$$f_k(\mathbf{h}) := \frac{1}{2} \sum_{i \in \mathscr{I}_k} w_i^{(k)} d_{Q_k}^2(\mathbf{h}, S_i), \tag{11}$$

where $(S_k)_{k \in \mathbb{N}}$ is a sequence of closed convex sets, $\mathscr{I}_k \subset \{0, 1, \ldots, k\}$ the indices of the closed convex sets and $w_i^{(k)} \in (0, 1]$, $i \in \mathscr{I}_k$ are the weights satisfying $\sum_{i \in \mathscr{I}_k} w_i^{(k)} = 1$. In this case, the update (10) can be expressed as

$$\mathbf{h}_{k+1} := \mathrm{prox}_{\beta_k g_k}^{(Q_k)}\left(\mathbf{h}_k - \beta_k \sum_{i \in \mathscr{I}_k} w_i^{(k)} \left(P_{S_i}^{(Q_k)}(\mathbf{h}_k) - \mathbf{h}_k\right)\right), \tag{12}$$

which is the generic form [17, 28] of an adaptive parallel projection type algorithm [26].

Example 4 (Weighted ℓ_1-norm for promoting sparsity) In order to exploit the sparsity of the unknown system, we can use, as φ_k or ψ_k in Algorithm 1,

$$\vartheta_k(\mathbf{h}) := \lambda \|\mathbf{h}\|_1^{(\omega_k)} := \lambda \sum_{i=1}^N \omega_i^{(k)} |h_i|, \tag{13}$$

where $\mathbf{h} := [h_1, h_2, \cdots, h_N]^T \in \mathbb{R}^N$, $\lambda > 0$ is the regularization parameter, and $\omega_i^{(k)} > 0$, $i \in \{1, 2, \ldots, N\}$, are the weights of the ℓ_1 norm. We restrict Q_k to a diagonal matrix, i.e., $Q_k := \mathrm{diag}\{q_1^{(k)}, q_2^{(k)}, \ldots, q_N^{(k)}\}$. Then the proximity operator of ϑ_k takes the form

$$\mathrm{prox}_{\gamma_k \vartheta_k}^{(Q_k)}(\mathbf{g}) = \sum_{i=1}^N \mathrm{sgn}(g_i) \max\left\{|g_i| - \frac{\gamma_k \lambda \omega_i^{(k)}}{q_i^{(k)}}, 0\right\} \mathbf{e}_i, \tag{14}$$

where $\{\mathbf{e}_i\}_{i=1}^N$ is the standard orthonormal basis of $\mathbb{R}^N$ (i.e., $\mathbf{e}_i := [0, \ldots, 0, 1, 0, \ldots, 0]^T$, $i \in \{1, 2, \ldots, N\}$, with the value 1 assigned to its ith position). We call the operator in (14) *Adaptively Weighted Soft-Thresholding (AWST)* [17, 28].

Now we turn our attention to the design of the weights $\omega_i^{(k)}$ in $\psi_k(\mathbf{h})$ in (13). The idea is to penalize coefficients that are close to zero with a large weight in order to push them down to zero. In addition, we will choose a small weight for active (large) coefficients in order to keep their influence onto the value of the cost function low (small contribution in (13)). Therefore, we will control the weight $\omega_i^{(k)}$ adaptively as a function of $h_i^{(k)}$ (the ith component of $\mathbf{h}_k$). One example of such a weight design in [17, 28] is

$$\omega_i^{(k)} := \begin{cases} \varepsilon, & \text{if } \left|h_i^{(k)}\right| > \tau, \\ 1, & \text{otherwise,} \end{cases} \tag{15}$$

where $\varepsilon \approx 0$ is a small positive constant, and $\tau > 0$ is the thresholding parameter for the selection of active coefficients.

There are many ways to design the parameter τ; for example, we may design the parameters based on noise statistics such as the variance.

Note that the computational complexity of AWST is relatively small, because it requires $\mathscr{O}(N)$ multiplications at most.

We mention that a different way of using a weighted ℓ_1-norm for sparse system identification was recently proposed [23], which is based on the use of the projection onto a weighted ℓ_1-norm constraint-set in the frame of the adaptive projected subgradient method [24, 25].

We finally note that our scheme can also utilize another sparsity-aware structure, for instance, the so-called *group-sparsity*, i.e., a coefficients in the same group are highly correlated and take on the values zero or non-zero as a group. This structure also often exhibit in many application. An adaptive filter exploiting the group-sparsity was recently proposed [2].

3.3 Proposed scheme for the sum of multiple convex functions

As a special example of our scheme, we present an online scheme for the minimization of a time-varying function which has a representation as the sum of several (more than two) convex functions.

The basic idea of our scheme is to reduce the minimization of the sum of (more than two) convex functions to the minimization of the sum of two functions in a product space: consider the situation where the time-varying cost function can be decomposed as the sum of convex functions

$$\Theta_k(\mathbf{h}) := \sum_{j=0}^J \theta_j^{(k)}(A_j \mathbf{h}), \tag{16}$$

where $\theta_j^{(k)} \colon \mathbb{R}^N \to (-\infty, \infty]$ is a proper lower semicontinuous convex function with an invertible matrix $A_j \in \mathbb{R}^{N \times N}$ ($j \in \{0, \ldots, J\} =: \mathscr{J}$). Let us define the product space $\mathscr{H} := \mathbb{R}^N \times \cdots \times \mathbb{R}^N$ ($J+1$ components). An element $H \in \mathscr{H}$ can be written as $H = (\mathbf{h}^{(0)}, \mathbf{h}^{(1)}, \ldots, \mathbf{h}^{(J)})$ with entries $\mathbf{h}^{(j)} \in \mathbb{R}^N$ $\forall j \in \mathscr{J}$. Using this notation it is obvious that

$$\mathbf{h}_k \in \arg\min_{\mathbf{h} \in \mathbb{R}^N} \Theta_k(\mathbf{h})$$

if and only if

$$\mathbf{j}_A(\mathbf{h}_k) \in \operatorname*{arg\,min}_{\mathbf{H}\in\mathscr{H}} \left(\Phi_k(\mathbf{H}) + \iota_D(\mathbf{H})\right),$$

where

$$\Phi_k \colon \mathscr{H} \to (-\infty,\infty],\ \mathbf{H} := (\mathbf{h}^{(0)},\mathbf{h}^{(1)},\dots,\mathbf{h}^{(J)}) \mapsto \sum_{j=0}^{J} \theta_j^{(k)}(\mathbf{h}^{(j)}),$$

$$\mathbf{j}_A \colon \mathbb{R}^N \to \mathscr{H}, \mathbf{h} \mapsto (A_0\mathbf{h}, A_1\mathbf{h},\dots,A_J\mathbf{h}) \in \mathscr{H},$$

and ι_D is the indicator function of the subspace $D := \{\mathbf{j}_A(\mathbf{h}) \in \mathscr{H} \mid \mathbf{h} \in \mathbb{R}^N\} \subset \mathscr{H}$. Hence the minimization of Θ_k in (16) can be reduced to the minimization of the sum of two functions $\Phi_k + \iota_D$ over the product space $\mathscr{H}$.

In addition, we have convenient forms of the proximity operator of Φ_k and ι_D for calculation by imposing the inner product $\langle\cdot,\cdot\rangle_{\mathscr{Q}_k} := \sum_{j=0}^{J} \nu_j \langle\cdot,\cdot\rangle_{Q_j^{(k)}}$ and its induced norm $\|\cdot\|_{\mathscr{Q}_k} := \sqrt{\sum_{j=0}^{J} \nu_j \|\cdot\|_{Q_j^{(k)}}^2}$ for adaptively-defined symmetric positive definite matrices $Q_j^{(k)} \subset \mathbb{R}^{N\times N}$ and positive weights $\nu_j \subset (0,\infty)$ ($j \in \mathscr{J}$) for the inner product. The components of the proximity operator of Φ_k (with respect to $\|\cdot\|_{\mathscr{Q}_k}$) are the respective proximity operator of θ_j^k, i.e.,

$$\operatorname{prox}_{\gamma_k\Phi_k}^{(\mathscr{Q}_k)}(\mathbf{H}) = \left(\operatorname{prox}_{\gamma_k\nu_0^{-1}\theta_0^{(k)}}^{(Q_0^{(k)})}(\mathbf{h}^{(0)}),\dots,\operatorname{prox}_{\gamma_k\nu_J^{-1}\theta_J^{(k)}}^{(Q_J^{(k)})}(\mathbf{h}^{(J)})\right)$$

for any $\mathbf{H} := (\mathbf{h}^{(0)},\mathbf{h}^{(1)},\dots,\mathbf{h}^{(J)})$. The components of $\operatorname{prox}_{\gamma_k\iota_D}^{(\mathscr{Q}_k)}(\mathbf{H})$ are a weighted average of the components of $\mathbf{H}$, i.e.,

$$\begin{aligned}\operatorname{prox}_{\gamma_k\iota_D}^{(\mathscr{Q}_k)}(\mathbf{H}) &= P_D^{(\mathscr{Q}_k)}(H)\\ &= \mathbf{j}_A\left(\left(\sum_{j=0}^{J} A_j^T Q_j^{(k)} A_j\right)^{-1}\left(\sum_{j=0}^{J} A_j^T Q_j^{(k)} \mathbf{h}^{(j)}\right)\right),\end{aligned}$$

for any $\mathbf{H} := (\mathbf{h}^{(0)},\mathbf{h}^{(1)},\dots,\mathbf{h}^{(J)})$.

Consequently, we arrive at the following algorithm that keeps the value of Θ_k low by applying Algorithm 1 to $\Phi_k + \iota_D$ with $\varphi_k := \Phi_k$ and $\psi_k := \iota_D$.

Algorithm 2 *For an arbitrarily chosen $\mathbf{g}_0 \in \mathbb{R}^N$, generate a sequence $\mathbf{h}_k \in \mathbb{R}^N$ ($k \in \mathbb{N}$) by*

$$\mathbf{h}_k := \left(\sum_{j=0}^{J} A_j^T Q_j^{(k)} A_j\right)^{-1}\left(\sum_{j=0}^{J} A_j^T Q_j^{(k)} \mathbf{g}_{k,j}\right) \tag{17}$$

with the sequences $\mathbf{g}_{k,j} \in \mathbb{R}^N$ ($k \in \mathbb{N}$) defined by $\mathbf{g}_{0,j} := \mathbf{g}_0$ and

$$\mathbf{g}_{k+1,j} := \mathbf{g}_{k,j} + t_k\left\{\operatorname{prox}_{\gamma_k\nu_j^{-1}\theta_j^{(k)}}^{(Q_j^{(k)})}\left(2A_j(\mathbf{h}_k) - \mathbf{g}_{k,j}\right) - A_j(\mathbf{h}_k)\right\} \tag{18}$$

for each $j \in \mathscr{J}$, where $\gamma_k \in (0,\infty)$ and $t_k \in (0,2)$ ($k \in \mathbb{N}$).

As we imposed $\psi_k = \iota_D$ in the derivation of Algorithm 2, Proposition 1(ii) is helpful to analyse the behavior of the algorithm.

Algorithm 2 covers many useful adaptive filtering algorithms because most convex cost functions adopted in existing algorithms [11, 15, 22] can be incorporated as $\theta_j^{(k)}(A_j\,\cdot\,)$ in Algorithm 2.

Example 5 (Sparsity-aware adaptive learning in transform domain) Assume that it is available that knowledge of the eigenvalue decomposition of the auto-correlation matrix of the input, i.e., $R_{\mathbf{u}} := E[\mathbf{u}_k\mathbf{u}_k^T] = V\Lambda V^T$ ($E[\cdot]$ denotes expectation). The auto-correlation in general degrades the convergence performance of adaptive filtering algorithms. For acceleration, we propose a learning algorithm for the transformed vector $\bar{\mathbf{h}}_k := \Lambda^{\frac{1}{2}} V\mathbf{h}_k$ (The bar "$\bar{\ }$" implies the coefficients in transformed domain). For simplicity, we restrict $Q_j^{(k)}$ to the identity matrix I.

Consider the time-varying cost function Θ_k in (16) in transform-domain with

$$\Theta_k(\bar{\mathbf{h}}) := \|V^T\Lambda^{-\frac{1}{2}}\bar{\mathbf{h}}\|_1^{(\omega_k)} + \sum_{j=1}^{J} \iota_{\bar{S}_{k+1-j}}(\bar{\mathbf{h}}) \tag{19}$$

Here, the first term represents the sparsity in the original domain and the set $\bar{S}_k := \left\{\bar{\mathbf{h}} \in \mathbb{R}^N \;\middle|\; \left|(\Lambda^{-\frac{1}{2}}V\mathbf{u}_k)^T\bar{\mathbf{h}} - d_k\right| \le \zeta_k\right\}$ represents the data-fidelity in the transform-domain.

To deal with (19), we set

$$\begin{aligned} A_0 &:= \Lambda^{-\frac{1}{2}}, & \theta_0^{(k)}(\bar{\mathbf{h}}) &:= \|V^T\bar{\mathbf{h}}\|_1^{(\omega_k)},\\ A_j &:= I, & \theta_j^{(k)}(\bar{\mathbf{h}}) &:= \iota_{\bar{S}_{k+1-j}}(\bar{\mathbf{h}}) \quad (j = 1,\dots,J)\end{aligned}$$

in the frame of (16). Then we obtain the following algorithm by direct application of Algorithm 2 (with the relation $\operatorname{prox}_{\gamma_k\|V^T(\cdot)\|_1^{(\omega_k)}}^{(I)} = V \circ \operatorname{prox}_{\gamma_k\|\cdot\|_1^{(\omega_k)}}^{(I)} \circ V^T$ [6]):

$$\begin{aligned}\mathbf{h}_k &= V^T\Lambda^{-\frac{1}{2}}\bar{\mathbf{h}}_k,\\ \bar{\mathbf{h}}_k &= \sum_{i=1}^{N}\left(\frac{(\lambda_i)^{-\frac{1}{2}}\bar{g}_{k,0,i} + \sum_{j=1}^{J}\bar{g}_{k,j,i}}{\lambda_i^{-1} + J}\right)\mathbf{e}_i,\\ \bar{\mathbf{g}}_{k+1,0} &= \bar{\mathbf{g}}_{k,0} + t_k\left\{V\operatorname{prox}_{\gamma_k\nu_0^{-1}\|\cdot\|_1^{(\omega_k)}}^{(I)}\left(2\mathbf{h}_k - V^T\bar{\mathbf{g}}_{k,0}\right) - \Lambda^{-\frac{1}{2}}\bar{\mathbf{h}}_k\right\},\\ \bar{\mathbf{g}}_{k+1,j} &= \bar{\mathbf{g}}_{k,j} + t_k\left\{P_{\bar{S}_{k+1-j}}^{(I)}\left(2\bar{\mathbf{h}}_k - \bar{\mathbf{g}}_{k,j}\right) - \bar{\mathbf{h}}_k\right\},\ (j = 1,\dots,J)\end{aligned}$$

where $g_{k,j,i}$ is the ith component of $\bar{\mathbf{g}}_{k,j}$, $\mathbf{e}_i$ is the standard orthonormal basis of $\mathbb{R}^N$, and the projection onto $\bar{S}_k$ is given by

$$P_{\bar{S}_k}^{(I)}(\bar{\mathbf{h}}) := \bar{\mathbf{h}} - \begin{cases} 0 & \text{if } \bar{\mathbf{h}} \in S_k,\\ t_k\dfrac{\varepsilon_k\left(V^T\Lambda^{-\frac{1}{2}}\bar{\mathbf{h}}\right) - \operatorname{sgn}\left(\varepsilon_k\left(V^T\Lambda^{-\frac{1}{2}}\bar{\mathbf{h}}\right)\right)\zeta_k}{\left\|\Lambda^{-\frac{1}{2}}V\mathbf{u}_k\right\|_I^2}\Lambda^{-\frac{1}{2}}V\mathbf{u}_k & \\ & \text{otherwise.}\end{cases}$$

4. NUMERICAL EXAMPLE

We examine the efficacy of Example 5 in the context of a simple echo cancellation problem for white noise input (i.e. $\bar{\mathbf{h}}_k = \mathbf{h}_k$).

We use the sparse echo impulse response $\mathbf{h}^*$ of length $N = 512$ initialized according to ITU-T G.168 [14] (in this case, $\mathbf{h}^*$ has only 64 non-zero components). The input signal u_k is generated according $\mathscr{N}(0,1)$. The noise n_k is zero mean white Gaussian and signal-to-noise ratio (SNR) $= 25$ dB, where $\text{SNR} := 10\log_{10}(E[z_k^2]/E[n_k^2])$ with $z_k := \mathbf{u}_k^T\mathbf{h}^*$.

For convenience, we denote by 'RZA-LMS' the Reweighted Zero-Attracting (RZA) LMS[4] [3], by 'APFBS' the adaptive proximal forward-backward splitting scheme [17] (i.e., the update (12)

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

$$\mathbf{h}_{k+1} := \mathbf{h}_k + \mu\frac{\varepsilon_k(\mathbf{h}_k)}{\|\mathbf{u}_k\|_2^2 + \delta}\mathbf{u}_k - \lambda\sum_{i=1}^{N}\frac{\operatorname{sgn}\left(h_i^{(k)}\right)}{1 + c_{\text{RZA}}\left|h_i^{(k)}\right|}\mathbf{e}_i,$$

where $r := 1$, $\mu > 0$ is the step-size and $c_{\text{RZA}} > 0$ is a constant. The regularization parameter $\delta \ge 0$ is introduced for numerical stability.

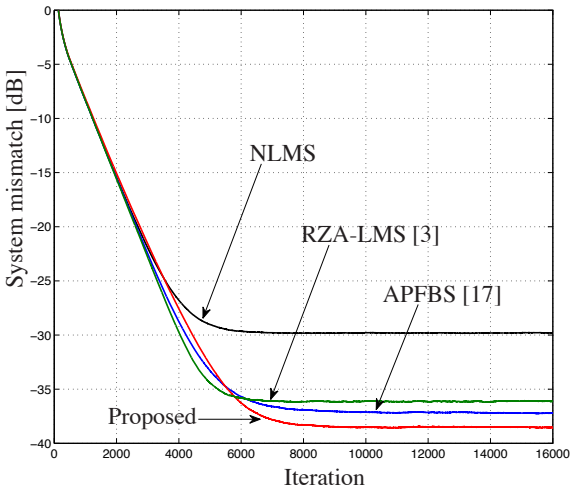

Figure 2: Comparison of the algorithms in system mismatch.

with S_k in (6) and $g_k := \lambda \|\cdot\|_1^{(\omega_k)}$ in (13)), and by 'Proposed' Example 5. 'NLMS' is nothing but APA (see (7)). We set $Q_k := I$, $r := 1$, $\varepsilon := 1 \times 10^{-6}$, and δ as the variance of the input signal. The parameters are set as $t_k := 0.5$ for 'NLMS', $(\mu, \lambda, c_{\mathrm{RZA}}) := (0.5,\ 10^{-4}, 2 \times 10^5)$ for 'RZA-LMS', $(\beta_k, \lambda, \tau) := (0.5,\ 8 \times 10^{-5}, 2 \times 10^{-4})$ for 'APFBS', and $(t_k, \nu_0, \tau, J, \zeta_k) := (1,\ 10^5, 5 \times 10^{-4}, 1, 5 \times 10^{-5})$ for 'Proposed'. For all the algorithms, we set the initial estimates to $\mathbf{g}_0 := \mathbf{0} \in \mathbb{R}^N$. The step-size for each algorithm is chosen in such a way that the convergence speed of all algorithms is the same. The regularization parameter is chosen to obtain the best results in our experiments.

Figure 2 depicts a comparison of the algorithms in the sense of system-mismatch $\eta(\mathbf{h}_k) := 10 \log_{10} \frac{\|\mathbf{h}^* - \mathbf{h}_k\|_2^2}{\|\mathbf{h}^*\|_2^2}$ averaged over 300 runs. 'Proposed' achieves the best steady-state behavior (lowest system mismatch) of the algorithms.

5. CONCLUSION

In this report, we extended the Douglus-Rachford splitting algorithm to the setting of a time-varying cost function. By this extension, we can deal with the sum of multiple time-varying nonsmooth convex functions for sparsity aware adaptive filtering. We presented some fundamental properties of the proposed scheme and many useful examples of the proposed scheme by specifying nonsmooth convex functions.

Acknowledgement

We would like to thank Mr. Naoto Fukuda (Tokyo Institute of Technology) for fruitful discussion.

REFERENCES

[1] J. Benesty, Y. A. Huang, J. Chen, and P. A. Naylor. *Adaptive algorithm for the identification of sparse impulse responses*, chapter 5, pages 125–153. Springer, 2006.

[2] Y. Chen, Y. Gu, and A. O. Hero. Regularized least-mean-square algorithms. Available at arxiv.org: http://arxiv.org/pdf/1012.5066.

[3] Y. Chen, Y. Gu, and A. O. Hero. Sparse LMS for system identification. In *Proc. IEEE ICASSP 2009*, pages 3125–3128, Taipel, Taiwan, Apr. 2009.

[4] P. L. Combettes and J.-C. Pesquet. A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE J. Sel. Top. Signal Process.*, 1:564–574, 2007.

[5] P. L. Combettes and J.-C. Pesquet. A proximal decomposition method for solving convex variational inverse problems. *Inverse Problems*, 24, 2008.

[6] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, 2005.

[7] D. L. Duttweiler. Proportionate normalized least-mean-squares adaptation in echo cancelers. *IEEE Trans. Speech Audio Processing*, 8(5):508–518, Sep. 2000.

[8] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2), 2011.

[9] S. Gandy and I. Yamada. Convex optimization techniques for the efficient recovery of a sparsely corrupted low-rank matrix. *Journal of Math-for-Industry*, JMI2010B, Oct. 2010.

[10] T. Gänsler, S. L. Gay, M. M. Sondhi, and J. Benesty. Double-talk robust fast converging algorithms for network echo cancellation. *IEEE Trans. Speech Audio Processing*, 8(6):656–663, Nov. 2000.

[11] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, third edition, 1996.

[12] T. Hinamoto and S. Maekawa. Extended theory of learning identification. *Trans. IEE Japan (in Japanese)*, 95-C(10):227–234, 1975.

[13] J. B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms*, volume 1 and 2. Springer, 1993.

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

[15] W. K. Jenkins and D. F. Marshall. Transform domain adaptive filtering. In *Digital Signal Processing Handbook*, chapter 22. CRC Press, Florida, 1999.

[16] J. J. Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. *C. R. Acad. Sci. Paris Sér*, 255:2897–2899, 1962.

[17] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada. A sparse adaptive filtering using time-varying soft-thresholding techniques. In *Proc. IEEE ICASSP 2010*, pages 3734–3737, Dallas, Texas, USA, Mar. 2010.

[18] J. Nagumo and A. Noda. A learning method for system identification. *IEEE Trans. Autom. Control*, 12(3):282–287, Jun. 1967.

[19] K. Ozeki and T. Umeda. An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties. *IEICE Trans. (in Japanese)*, 67-A(5):126–132, 1984.

[20] B. G. Passty. Ergodic convergence to a zero of the sum of monotone operators in hilbert space. *J. Math. Anal. Appl.*, 72(2):383–390, 1979.

[21] G. Pierra. Decomposition through formalization in a product space. *Math. Program.*, 28:96–115, 1984.

[22] A. H. Sayed. *Fundamentals of Adaptive Filtering*. Wiley, 2003.

[23] K. Slavakis, Y. Kopsinis, and S. Theodoridis. Adaptive algorithm for sparse system identification using projections onto weighted l1 balls. In *Proc. IEEE ICASSP 2010*, pages 3742–3745, Dalla, Texas, USA, Mar. 2010.

[24] 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 Processing Magazine*, 28(1):97–123, Jan. 2011.

[25] I. Yamada and N. Ogura. Adaptive projected subgradient method for asymptotic minimization of sequence of nonnegative convex functions. *Numer. Funct. Anal. Optim.*, 25(7&8):593–617, 2004.

[26] I. Yamada, K. Slavakis, and K. Yamada. An efficient robust adaptive filtering algorithm based on parallel subgradient projection techniques. *IEEE Trans. Signal Processing*, 50(5):1091–1101, 2002.

[27] 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* (H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, eds.), chapter 17, pages 345–390. Springer, 2011.

[28] 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*, Biopolis, Singapore, Dec. 2010.

[29] 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*, Prague, Czech Republic, May. 2011.

[30] M. Yukawa, K. Slavakis, and I. Yamada. Adaptive parallel quadratic-metric projection algorithms. *IEEE Trans. Audio, Speech, Language Processing*, 15(5):1665–1680, 2007.