KRYLOV-PROPORTIONATE NLMS ALGORITHM BASED ON MULTISTAGE WIENER FILTER REPRESENTATION

Masahiro Yukawa

Next Generation Mobile Communications Laboratory, RIKEN, JAPAN

ABSTRACT

This paper proposes a fast converging adaptive filtering algorithm named *Krylov-proportionate normalized least mean-square (KPNLMS)* by extending *the proportionate normalized least mean square (PNLMS)* algorithm. PNLMS is known to exhibit faster convergence than the standard NLMS algorithm for sparse unknown systems. The proposed algorithm attains similar effects for non-sparse unknown systems by constructing, based on the *multistage Wiener filter (MWF)* representation, an orthonormal basis with which the unknown system has a sparse structure. The proposed algorithm can be analyzed by the *adaptive parallel variable-metric projection* framework. Numerical studies for non-sparse unknown systems are presented, comparing KPNLMS and the MWF-based reduced-rank method.

Index Terms— Adaptive filters, multistage Wiener filter, proportionate NLMS

1. PROBLEM FORMULATION

We consider a simple linear system model:

$$d_k := \boldsymbol{u}_k^T \boldsymbol{h}^* + n_k,\ k \in \mathbb{N}, \tag{1}$$

where $\boldsymbol{u}_k := [u_k, u_{k-1}, \cdots, u_{k-N+1}]^T \in \mathbb{R}^N$ is the input vector at time k with the input process $(u_k)_{k\in\mathbb{N}}$, $\boldsymbol{h}^* \in \mathbb{R}^N$ the estimandum (an unknown linear system), $(d_k)_{k\in\mathbb{N}}$ the output process, and $(n_k)_{k\in\mathbb{N}}$ the noise process. The available is the input and output processes, and an adaptive filter $(\boldsymbol{h}_k)_{k\in\mathbb{N}}$ is controlled in a recursive way by an adaptive algorithm to estimate $\boldsymbol{h}^*$. A common criterion for the problem is the mean-square error (MSE):

$$\mathrm{MSE}(\boldsymbol{h}) := E\left\{\left(d_k - \boldsymbol{u}_k^T \boldsymbol{h}\right)^2\right\},\ \boldsymbol{h} \in \mathbb{R}^N, \tag{2}$$

where $E\{\cdot\}$ denotes expectation. The filter $\boldsymbol{h}_{\mathrm{MMSE}} \in \mathbb{R}^N$ minimizing (2) is called the minimum mean-square error (MMSE) filter, characterized by the normal equation widely known as *the Wiener-Hopf equation*: $\boldsymbol{R}\boldsymbol{h}_{\mathrm{MMSE}} = \boldsymbol{p}$, where $\boldsymbol{R} := E\{\boldsymbol{u}_k \boldsymbol{u}_k^T\} \in \mathbb{R}^{N\times N}$ and $\boldsymbol{p} := E\{\boldsymbol{u}_k d_k\} \in \mathbb{R}^N$. The autocorrelation matrix $\boldsymbol{R}$ is mostly positive definite due to the presence of noise, and in this case, the MMSE filter is uniquely given by $\boldsymbol{h}_{\mathrm{MMSE}} = \boldsymbol{R}^{-1}\boldsymbol{p}$.

2. PROPOSED ADAPTIVE FILTERING TECHNIQUES

The proposed techniques are constructed by the following three procedures:

1) whiten the input process $(u_k)_{k\in\mathbb{N}}$;
2) construct an orthonormal basis set with which the estimandum is sparse;
3) extract and exploit the sparse structure with no prior information.

The main procedures are 2) and 3), and the procedure 1) is an idea to help the estimandum be sparse in the coordinate obtained through the procedure 2). In case that the correlation of the input is weak, the estimandum tends to have a sparse structure in the coordinate constructed, and in such a case, the procedure 1) can be skipped. For this reason, we present a scheme composed only of procedures 2) and 3) due to the lack of space.

2.1. Krylov-Proportionate NLMS Algorithm

In the multistage Wiener filter (MWF) representation [1], the estimandum $\boldsymbol{h}^*$ is expressed with the set of basis vectors whose matrix-form expression is given by

$$\begin{aligned}\boldsymbol{L}_N &:= [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \cdots, \boldsymbol{\ell}_N] \\ &:= \left[\boldsymbol{v}_1, \boldsymbol{B}_1^T\boldsymbol{v}_2, \cdots \prod_{i=N-2}^{1} \boldsymbol{B}_i^T \boldsymbol{v}_{N-1},\ \prod_{i=N-1}^{1} \boldsymbol{B}_i^T\right] \in \mathbb{R}^{N\times N}.\end{aligned}$$

Here, $\boldsymbol{v}_1 := \boldsymbol{p}/\|\boldsymbol{p}\| \in \mathbb{R}^N$, where $\|\cdot\|$ denotes the 2-norm, $\boldsymbol{B}_1 \in \mathbb{R}^{(N-1)\times N}$ is *the blocking matrix* satisfying $\boldsymbol{B_1 v}_1 = \boldsymbol{0}$, $\boldsymbol{v}_2 := \boldsymbol{p}_2/\|\boldsymbol{p}_2\| \in \mathbb{R}^{N-1}$ with $\boldsymbol{p}_2 := E\{\boldsymbol{u}_k^{(1)} d_k^{(1)}\} = \boldsymbol{B}_1\boldsymbol{R}\boldsymbol{v}_1 \in \mathbb{R}^{N-1}$, where $\boldsymbol{u}_k^{(1)} := \boldsymbol{B}_1\boldsymbol{u}_k \in \mathbb{R}^{N-1}$ and $d_k^{(1)} := \boldsymbol{u}_k^T\boldsymbol{v}_1 \in \mathbb{R}$. $\boldsymbol{B}_i$ and $\boldsymbol{v}_i$ for $i = 2, 3, \cdots, N-1$ are defined in the same manner. In [2], it is shown that the subspace $\mathcal{S}_D := \mathcal{R}\left\{[\boldsymbol{L}_N]_{1:N,1:D}\right\} \subset \mathbb{R}^N$ is equivalent to the Dth Krylov subspace $\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p}) := \mathcal{R}\{\boldsymbol{K}_D(\boldsymbol{R},\boldsymbol{p})\} \subset \mathbb{R}^N$. Here, $\boldsymbol{K}_D(\boldsymbol{R},\boldsymbol{p}) := \left[\boldsymbol{p}, \boldsymbol{R}\boldsymbol{p}, \boldsymbol{R}^2\boldsymbol{p}, \cdots, \boldsymbol{R}^{D-1}\boldsymbol{p}\right] \in \mathbb{R}^{N\times D}$; $\mathcal{R}\{\cdot\}$ denotes *range* (column space); and $[\boldsymbol{A}]_{a:b,c:d}$ for a given matrix $\boldsymbol{A}$ designates its sub-matrix corresponding to the ath to bth rows and the cth to dth columns.

The reduced-rank adaptive filter based on MWF stems from the idea that the optimal MMSE filter $\boldsymbol{h}_{\mathrm{MMSE}}$ should be approximated by a vector in $\mathcal{K}_D(\boldsymbol{R},\boldsymbol{p})$ [typically $4 \le D \le 8$]. This suggests that, if we express $\boldsymbol{h}_{\mathrm{MMSE}}$ with the matrix of the MWF basis set as $\boldsymbol{h}_{\mathrm{MMSE}} = \boldsymbol{L}_N\boldsymbol{z}$, the coefficient vector $\boldsymbol{z} \in \mathbb{R}^N$ is expected to be *sparse*; specifically only first few entries of $\boldsymbol{h}_{\mathrm{MMSE}}$ would have large magnitudes. Instead of using the MWF matrix $\boldsymbol{L}_N$, we use the following simplified orthogonal matrix for computational saving:

$$\boldsymbol{U} = [\boldsymbol{S}\ \boldsymbol{W}] \in \mathbb{R}^{N\times N}. \tag{3}$$

Here, $\boldsymbol{S} \in \mathbb{R}^{N\times D}$ is constructed by orthogonalizing the columns of $\boldsymbol{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})$ associated with an estimated autocorrelation matrix $\widehat{\boldsymbol{R}} \in \mathbb{R}^{N\times N}$ and an estimated cross-correlation vector $\widehat{\boldsymbol{p}} \in \mathbb{R}^N$; and $\boldsymbol{W} \in \mathbb{R}^{N\times(N-D)}$ can be chosen arbitrarily. Note that, if the input is completely white, i.e., $\boldsymbol{R} = \sigma^2\boldsymbol{I}$ for some $\sigma^2 > 0$ ($\boldsymbol{I}$ denotes the identity matrix), then the dimension of $\boldsymbol{K}_D(\widehat{\boldsymbol{R}},\widehat{\boldsymbol{p}})$ is one, and its basis vector is the MMSE filter $\boldsymbol{h}_{\mathrm{MMSE}}$ itself. This

strongly suggests that $\boldsymbol{h}_{\mathrm{MMSE}}$ should have a sparse structure in the coordinate associated with the columns of the matrix $\boldsymbol{U}$ especially when the input signal has fairly weak correlation. Namely, the vector $\bar{\boldsymbol{h}}_{\mathrm{MMSE}} := \boldsymbol{U}^T\boldsymbol{h}_{\mathrm{MMSE}} \in \mathbb{R}^N$ is highly expected to be sparse.

The next step is extraction of the sparseness in the coordinate associated with the columns of the matrix $\boldsymbol{U}$. The strategy is to extend the idea of the proportionate normalized least mean-square (PNLMS) algorithm [3]; instead of the original version, we adopt the improved PNLMS (IPNLMS) algorithm proposed in [4]. The sparseness of $\bar{\boldsymbol{h}}_{\mathrm{MMSE}}$ will be reflected in $\bar{\boldsymbol{h}}_k := \boldsymbol{U}^T\boldsymbol{h}_k \in \mathbb{R}^N$. We thus construct a diagonal matrix adjusting the step size in the direction of each basis vector as

$$\boldsymbol{\Lambda}_k := \mathrm{diag}\{\theta_1^{(k)}, \theta_2^{(k)}, \cdots, \theta_N^{(k)}\} \in \mathbb{R}^{N\times N},\ k \in \mathbb{N}, \tag{4}$$

with

$$\theta_n^{(k)} := \frac{1-\eta}{N} + \eta\frac{\left|\left[\bar{\boldsymbol{h}}_k\right]_n\right|}{\left\|\bar{\boldsymbol{h}}_k\right\|_1 + \varepsilon},\ n = 1, 2, \cdots, N,\ k \in \mathbb{N}. \tag{5}$$

Here, $|\cdot|$ and $\|\cdot\|_1$ denote absolute value and 1-norm, respectively, $\eta \in (0,1)$ a factor to control the amount of proportionality in the update, and $\varepsilon > 0$ a small positive constant for regularization. The update equation of the adaptive filter is given as follows:

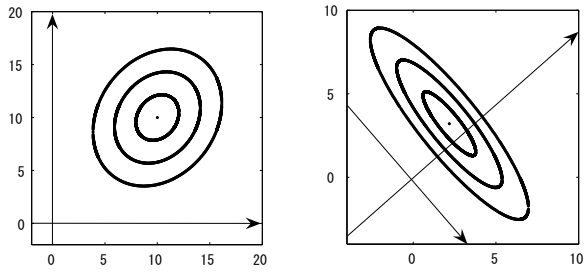
$$\boldsymbol{h}_{k+1} = \boldsymbol{h}_k - \lambda_k e_k(\boldsymbol{h}_k)\frac{\boldsymbol{\Omega}_k\boldsymbol{u}_k}{\boldsymbol{u}_k^T\boldsymbol{\Omega}_k\boldsymbol{u}_k},\ k \in \mathbb{N}, \tag{6}$$

where $\boldsymbol{\Omega}_k := \boldsymbol{U}\boldsymbol{\Lambda}_k\boldsymbol{U}^T$ is a positive definite matrix, $e_k : \mathbb{R}^N \to \mathbb{R}$, $\boldsymbol{h} \mapsto \boldsymbol{u}_k^T\boldsymbol{h} - d_k$, is the error function at kth iteration, and $\lambda_k \in [0,2]$ the step size. If $\boldsymbol{U} = \boldsymbol{I}$, the equation in (6) is nothing but the IPNLMS algorithm, thus it is a natural extention of IPNLMS (Note: $\eta = (\alpha+1)/2$ for the α used in [4]). Moreover, the equation in (6) is a special case of *the adaptive parallel variable-metric projection (APVP) algorithm* [5] with $q = 1$, $\boldsymbol{Q}_k = \boldsymbol{\Omega}_k^{-1}$, $T_k = I$ (I denotes the identity mapping), and $C_1^{(k)} = \{\boldsymbol{v} \in \mathbb{R}^N : e_k(\boldsymbol{v}) = 0\}$. Given $K_0 \in \mathbb{N}$, which specifies the training period for estimating $\boldsymbol{R}$ and $\boldsymbol{p}$, the proposed scheme is summarized below.

Scheme 1 (Krylov-Proportionate NLMS Algorithm)
Given any initial estimates $\widehat{\boldsymbol{R}}_0 \in \mathbb{R}^{N\times N}$ and $\widehat{\boldsymbol{p}}_0 \in \mathbb{R}^N$, generate a sequence $(\boldsymbol{h}_k)_{k\in\mathbb{N}} \subset \mathbb{R}^N$ with $\boldsymbol{h}_0 = \boldsymbol{0}$ recursively as follows.
if $0 \le k < K_0$
  $\widehat{\boldsymbol{R}}_{k+1} := \gamma\widehat{\boldsymbol{R}}_k + (1-\gamma)\boldsymbol{u}_k\boldsymbol{u}_k^T$ *($\gamma \in [0,1)$: the forgetting factor)*
  $\widehat{\boldsymbol{p}}_{k+1} := \gamma\widehat{\boldsymbol{p}}_k + (1-\gamma)\boldsymbol{u}_k d_k$
  $\boldsymbol{\Omega}_k := \boldsymbol{I}$
else
  if $k = K_0$
    $\boldsymbol{G} := \boldsymbol{K}_D(\widehat{\boldsymbol{R}}_{k+1}, \widehat{\boldsymbol{p}}_{k+1}) \in \mathbb{R}^{N\times D}$
    $\boldsymbol{U} \in \mathbb{R}^{N\times N}$ *: orthonormalized version of* $[\boldsymbol{G}\ \boldsymbol{H}]$
      ($\boldsymbol{H} \in \mathbb{R}^{N\times(N-D)}$: a random matrix)
  end;
  $\boldsymbol{\Omega}_k := \boldsymbol{U}\boldsymbol{\Lambda}_k\boldsymbol{U}^T$ *with $\boldsymbol{\Lambda}_k$ given in* (4)
end;
  Filter update: see (6)

Scheme 1 utilizes (i) the orthogonal matrix $\boldsymbol{U}$ constructed with the Krylov sequence, and (ii) the diagonal matrix $\boldsymbol{\Lambda}_k$ based on the idea of PNLMS, thus named *Krylov-Proportionate Normalized Least Mean-Square (KPNLMS) Algorithm*. We remark that the proposed scheme can be generalized with parallel projection in the framework of APVP [5]; the convergence analysis of APVP has been presented in [6] with the aid of *the adaptive projected subgradient method* [7]. The analysis in [6] can be applied straightforwardly to Scheme 1. It should be remarked that, if the input is nonstationary process, then the computation of $\widehat{\boldsymbol{R}}_k$ and $\widehat{\boldsymbol{p}}_k$ should be performed continuously (or periodically), based on which the matrix $\boldsymbol{U}$ should be recomputed periodically.

The matrix-vector multiplication in the filter update involves $O(N^2)$ multiplications, which would not be attractive from an implementation point of view in some applications. Therefore, a simplified scheme with only $O(N)$ complexity is presented in Sec. 2.3.



(a) Original (b) Proposed

Fig. 1. Impact of the proposed transformation $\boldsymbol{u}_k \mapsto \boldsymbol{v}_k := \boldsymbol{\Omega}_k^{1/2}\boldsymbol{u}_k$ on the error surface contours.

2.2. Error Surface Analysis

To perform an error surface analysis, we left-multiply both-sides of (6) by $\boldsymbol{\Omega}_k^{-1/2}$, yielding

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \lambda_k\frac{{\boldsymbol{w}_k}^T\boldsymbol{v}_k - d_k}{\|\boldsymbol{v}_k\|^2}\boldsymbol{v}_k,\ k \in \mathbb{N}, \tag{7}$$

where $\boldsymbol{w}_k := \boldsymbol{\Omega}_k^{-1/2}\boldsymbol{h}_k$ and $\boldsymbol{v}_k := \boldsymbol{\Omega}_k^{1/2}\boldsymbol{u}_k$. The MMSE solution is modified into $\boldsymbol{w}_{\mathrm{MMSE}}$ characterized by $E\{\boldsymbol{v}_k\boldsymbol{v}_k^T\}\boldsymbol{w}_{\mathrm{MMSE}} := E\{\boldsymbol{v}_k d_k\}$.

Suppose for simplicity that the input is uncorrelated, i.e., the equal-error contours are hyperspheres. Then, left-multiplying $\boldsymbol{u}_k$ by $\boldsymbol{\Omega}_k^{1/2}(=\boldsymbol{U}\boldsymbol{\Lambda}_k^{1/2}\boldsymbol{U}^T)$ modifies the contours into hyperellipse whose axes are the column vectors of $\boldsymbol{U}$ and whose radiuses are proportionate inversely to the diagonal elements of $\boldsymbol{\Lambda}_k^{1/2}$. An example is illustrated in Fig. 1: Figs. 1(a) and 1(b) draw the equal-error contours of the MSE surfaces for (a) the standard NLMS algorithm and (b) Scheme 1, respectively. In the figure, it is assumed that $N = 2$, $\boldsymbol{h}_{\mathrm{MMSE}} = [10, 10]^T$, and $\boldsymbol{R} = \begin{bmatrix} 1, & -0.2 \\ -0.2 & 0.9 \end{bmatrix}$. In this case, $\boldsymbol{U} \approx \begin{bmatrix} 0.7526 & 0.6585 \\ 0.6585 & -0.7526 \end{bmatrix}$, and $\bar{\boldsymbol{h}}_{\mathrm{MMSE}} \approx [14.11, -0.9407]^T$ is sparse. For Scheme 1, we set $\boldsymbol{\Lambda}_k = \mathrm{diag}\left\{\left|\bar{\boldsymbol{h}}_{\mathrm{MMSE}}\right|\right\}$, where $|\cdot|$ for a vector argument denotes elementwise absolute-value.

Consider the shape of the surface of the MSE function for Scheme 1 with reference to Fig. 1(b). The minimum point (or center) is $\boldsymbol{w}_{\mathrm{MMSE}}$, and, looking from the origin, the slope is 'steep' in the direction of an axis in which $\boldsymbol{w}_{\mathrm{MMSE}}$ has a large magnitude. This suggests that Scheme 1 steps larger than the standard NLMS algorithm, leading to fast convergence.

2.3. Simplification and Computational Complexity

The process to construct the orthogonal matrix $\boldsymbol{U}$ involves the computation for $\widehat{\boldsymbol{R}}_{k+1}$ and $\widehat{\boldsymbol{p}}_{k+1}$ during a short training-period, and the computation to construct and orthonormalize the matrix $[\boldsymbol{G}\ \boldsymbol{H}]$ only once. Hence we neglect such computation.

The main computational burden per iteration is involved with the construction of the diagonal matrix $\boldsymbol{\Lambda}_k$ and the filter update. The computational complexity of the filter update would be $O(N^2)$ with the matrix $\boldsymbol{\Omega}_k$ having no special structure; see [5]. Fortunately, however, the replacement of $\boldsymbol{\Lambda}_k$ by the following simplified matrix leads to $O(N)$ complexity with little degradation of performance:

$$\boldsymbol{\Lambda}_k^{(2)} := \mathrm{diag}\{\widetilde{\theta}_1^{(k)}, \cdots, \widetilde{\theta}_D^{(k)}, \delta_k, \cdots, \delta_k\},\ k \in \mathbb{N}, \tag{8}$$

where

$$\delta_k := \frac{1-\eta}{N} + \eta \frac{\varsigma_k \left\| \left[\bar{\boldsymbol{h}}_k\right]_{1:D} \right\|_1 / D}{\psi_k + \varepsilon},$$
$$\varsigma_k > 0,\ \psi_k := [1 + \varsigma_k (N-D)/D] \left\| \left[\bar{\boldsymbol{h}}_k\right]_{1:D} \right\|_1, \tag{9}$$

$$\widetilde{\theta}_n^{(k)} := \frac{1-\eta}{N} + \eta \frac{\left|\left[\bar{\boldsymbol{h}}_k\right]_n\right|}{\psi_k + \varepsilon},\ n = 1, 2, \cdots, D. \tag{10}$$

The parameter ς_k is designed according to the energy relations of $\bar{\boldsymbol{h}}_k$ in $\mathcal{K}_D(\widehat{\boldsymbol{R}}_{k+1}, \widehat{\boldsymbol{p}}_{k+1})$ and its orthogonal compliment (cf. Sec. 2.4).

The computational complexity for $\boldsymbol{\Lambda}_k^{(2)}$ is approximately $D(< N)$. Let $\boldsymbol{U} =: [\boldsymbol{U}_1\ \boldsymbol{U}_2]$ with $\boldsymbol{U}_1 \in \mathbb{R}^{N\times D}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{N\times(N-D)}$ and $\boldsymbol{\Lambda}_{k,D}^{(2)} := \mathrm{diag}\{\widetilde{\theta}_1^{(k)}, \cdots, \widetilde{\theta}_D^{(k)}\} \in \mathbb{R}^{D\times D}$. Then, defining $\boldsymbol{\Omega}_k^{(2)} := \boldsymbol{U}\boldsymbol{\Lambda}_k^{(2)}\boldsymbol{U}^T$, the portion $\boldsymbol{\Omega}_k^{(2)}\boldsymbol{u}_k$ involved with the filter update [cf. (6)] is computed as

$$\begin{aligned}\boldsymbol{\Omega}_k^{(2)}\boldsymbol{u}_k &= [\boldsymbol{U}_1\ \boldsymbol{U}_2] \begin{bmatrix} \boldsymbol{\Lambda}_{k,D}^{(2)} & \boldsymbol{O} \\ \boldsymbol{O} & \delta_k \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_1^T \\ \boldsymbol{U}_2^T \end{bmatrix} \boldsymbol{u}_k \\ &= \boldsymbol{U}_1 \left[\boldsymbol{\Lambda}_{k,D}^{(2)} \boldsymbol{U}_1^T \boldsymbol{u}_k - \delta_k \boldsymbol{U}_1^T \boldsymbol{u}_k \right] + \delta_k \boldsymbol{u}_k, \end{aligned} \tag{11}$$

where the last equality is verified by $\boldsymbol{U}_1\boldsymbol{U}_1^T + \boldsymbol{U}_2\boldsymbol{U}_2^T (= \boldsymbol{U}\boldsymbol{U}^T) = \boldsymbol{I}$. From (11), it is seen that $\boldsymbol{\Omega}_k^{(2)}\boldsymbol{u}_k$ requires only $(2D+1)N + 2D$ multiplications (we mostly have $D \ll N$).

Left-multiplying the both-sides of the filter update by $\boldsymbol{U}^T$, we obtain the following recursive form:

$$\bar{\boldsymbol{h}}_{k+1} = \bar{\boldsymbol{h}}_k - \lambda_k e_k(\boldsymbol{h}_k) \frac{\boldsymbol{\Lambda}_k^{(2)} \boldsymbol{U}^T \boldsymbol{u}_k}{\boldsymbol{u}_k^T \boldsymbol{\Omega}_k^{(2)} \boldsymbol{u}_k},\ k \in \mathbb{N}. \tag{12}$$

Since $\boldsymbol{u}_k^T \boldsymbol{\Omega}_k^{(2)} \boldsymbol{u}_k$ is obtainable through the process of filter update, the additional part to consider for updating $\bar{\boldsymbol{h}}_k$ is

$$\boldsymbol{\Lambda}_k^{(2)} \boldsymbol{U}^T \boldsymbol{u}_k = \begin{bmatrix} \boldsymbol{\Lambda}_{k,D}^{(2)} \boldsymbol{U}_1^T \boldsymbol{u}_k \\ \delta_k \boldsymbol{U}_2^T \boldsymbol{u}_k \end{bmatrix}. \tag{13}$$

Fortunately, from (8)–(10), the matrix $\boldsymbol{\Lambda}_k^{(2)}$ depends only on the subvector $[\bar{\boldsymbol{h}}_{k+1}]_{1:D}$, which is updated recursively as

$$\left[\bar{\boldsymbol{h}}_{k+1}\right]_{1:D} = \left[\bar{\boldsymbol{h}}_k\right]_{1:D} - \lambda_k e_k(\boldsymbol{h}_k) \frac{\boldsymbol{\Lambda}_{k,D}^{(2)} \boldsymbol{U}_1^T \boldsymbol{u}_k}{\boldsymbol{u}_k^T \boldsymbol{\Omega}_k^{(2)} \boldsymbol{u}_k}. \tag{14}$$

Since $\boldsymbol{\Lambda}_{k,D}^{(2)} \boldsymbol{U}_1^T \boldsymbol{u}_k$ is obtainable through the process of filter update [see (11)], the update of $[\bar{\boldsymbol{h}}_k]_{1:D}$ requires only $D+1$ additional multiplications.

Consequently, the simplified scheme is obtained by replacing $\boldsymbol{\Omega}_k$ in Scheme 1 by $\boldsymbol{\Omega}_k^{(2)}$ with (8)–(11) and (14). The overall computational complexity of the simplified scheme, including the design of the matrix $\boldsymbol{\Lambda}_k^{(2)}$, is $(2D+3)N + 4D$ at each iteration.

2.4. Energy of $\boldsymbol{h}_{\mathrm{MMSE}}$ in the Krylov Subspace $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$

Scheme 1 is expected to be effective especially when the MMSE solution $\boldsymbol{h}_{\mathrm{MMSE}}$ has large energy in the Krylov subspace $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ for a small D. This section is devoted to examine how the autocorrelation matrix $\boldsymbol{R} \in \mathbb{R}^{N\times N}$ and the cross-correlation vector $\boldsymbol{p} \in \mathbb{R}^N$ is related to the energy. To quantify the energy, we define *the normalized energy* of an arbitrary vector $\boldsymbol{a} \in \mathbb{R}^N$ in a subspace $\mathcal{M} \subset \mathbb{R}^N$ as follows:

$$\mathrm{Ener}(\boldsymbol{a}, \mathcal{M}) := \frac{\|P_{\mathcal{M}}(\boldsymbol{a})\|^2}{\|\boldsymbol{a}\|^2} \in [0, 1]. \tag{15}$$

Here, $P_{\mathcal{M}}(\boldsymbol{a})$ stands for the metric projection of $\boldsymbol{a}$ onto $\mathcal{M}$, determined uniquely and characterized by $\|\boldsymbol{a} - P_{\mathcal{M}}(\boldsymbol{a})\| = \min_{\boldsymbol{b}\in\mathcal{M}} \|\boldsymbol{a} - \boldsymbol{b}\|$. Let $\boldsymbol{R} = \boldsymbol{U}_{\boldsymbol{R}}\boldsymbol{\Lambda}_{\boldsymbol{R}}\boldsymbol{U}_{\boldsymbol{R}}^T$ be the eigenvalue decomposition of $\boldsymbol{R}$ with an orthogonal matrix $\boldsymbol{U}_{\boldsymbol{R}}$ and a diagonal matrix $\boldsymbol{\Lambda}_{\boldsymbol{R}} := \mathrm{diag}\{\theta_1, \theta_2, \cdots, \theta_N\}$, where $\theta_1, \theta_2, \cdots, \theta_N$ denote the eigenvalues of $\boldsymbol{R}$ corresponding to the column vectors of $\boldsymbol{U}_{\boldsymbol{R}}$. The simplified notation $\boldsymbol{K}_D$ is used to express $\boldsymbol{K}_D(\boldsymbol{R}, \boldsymbol{p})$ $(:= [\boldsymbol{p}, \boldsymbol{R}\boldsymbol{p}, \boldsymbol{R}^2\boldsymbol{p}, \cdots, \boldsymbol{R}^{D-1}\boldsymbol{p}]) \in \mathbb{R}^{N\times D}$. The energy of $\boldsymbol{h}_{\mathrm{MMSE}}$ in $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ can then be expressed as follows (all the results are given with no proof in this section for lack of space).

Proposition 1 *Given $\boldsymbol{R} \neq \boldsymbol{O}$ and $\boldsymbol{p} \neq \boldsymbol{0}$, the energy of $\boldsymbol{h}_{\mathrm{MMSE}}$ in $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ is given as*

$$\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})) = \frac{\left\|P_{\mathcal{R}\{\boldsymbol{\Sigma}\}}(\boldsymbol{\pi})\right\|^2}{\|\boldsymbol{\pi}\|^2}, \tag{16}$$

where $\boldsymbol{\pi} := \boldsymbol{U}_{\boldsymbol{R}}^T \boldsymbol{h}_{\mathrm{MMSE}} \in \mathbb{R}^N$ and $\boldsymbol{\Sigma} := \boldsymbol{U}_{\boldsymbol{R}}^T \boldsymbol{K}_D(\boldsymbol{R}, \boldsymbol{p}) = \left[\boldsymbol{\Lambda}_{\boldsymbol{R}}\boldsymbol{\pi}, \boldsymbol{\Lambda}_{\boldsymbol{R}}^2\boldsymbol{\pi}, \cdots, \boldsymbol{\Lambda}_{\boldsymbol{R}}^D\boldsymbol{\pi}\right] \in \mathbb{R}^{N\times D}$.

$P_{\mathcal{R}\{\boldsymbol{\Sigma}\}}(\boldsymbol{\pi})$ has a closed-form expression $P_{\mathcal{R}\{\boldsymbol{\Sigma}\}}(\boldsymbol{\pi}) = \boldsymbol{\Sigma}\boldsymbol{\Sigma}^\dagger\boldsymbol{\pi}$, where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse. In particular case that $\boldsymbol{K}_D(\boldsymbol{R}, \boldsymbol{p})$ has full-column rank so $\boldsymbol{\Sigma}$ does, $P_{\mathcal{R}\{\boldsymbol{\Sigma}\}}(\boldsymbol{\pi}) = \boldsymbol{\Sigma}(\boldsymbol{\Sigma}^T\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}^T\boldsymbol{\pi}$.

From Proposition 1, it is readily verified that

$$\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\}) = \left\langle \frac{\boldsymbol{\pi}}{\|\boldsymbol{\pi}\|}, \frac{\boldsymbol{\Lambda}_{\boldsymbol{R}}\boldsymbol{\pi}}{\|\boldsymbol{\Lambda}_{\boldsymbol{R}}\boldsymbol{\pi}\|} \right\rangle^2, \tag{17}$$

where $\boldsymbol{s}_1 := \boldsymbol{p}/\|\boldsymbol{p}\| \in \mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$ for $D \geq 1$. The equation (17) suggests the following. In case that $\boldsymbol{p}$ is an eigenvector of $\boldsymbol{R}$ ($\Leftrightarrow \boldsymbol{h}_{\mathrm{MMSE}} = \alpha\boldsymbol{p}, \exists\alpha \in \mathbb{R}$), $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\}) = 1$. In case that $\boldsymbol{p}$ is *not* an eigenvector of $\boldsymbol{R}$, $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\})$ depends on the eigenvalue-spread of $\boldsymbol{R}$. If the input signal is weakly correlated, then the 'angle' between $\boldsymbol{\pi}$ and $\boldsymbol{\Lambda}_{\boldsymbol{R}}\boldsymbol{\pi}$ is small due to small eigenvalue-spread, implying that the energy $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\})$ is close fairly to unity. Conversely, if the input signal is strongly correlated, $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\})$ would be far from unity.

In case that $\boldsymbol{p}$ is *not* an eigenvector of $\boldsymbol{R}$, how is the energy compensated in the other basis vectors of $\mathcal{K}_D(\boldsymbol{R}, \boldsymbol{p})$? We see from Proposition 1 that it depends on the angle between $\boldsymbol{\pi}$ and $\boldsymbol{\Lambda}_{\boldsymbol{R}}^d\boldsymbol{\pi}$ $(d = 2, \cdots, D)$. We further examine the energy in the second orthonormal-basis vector $\boldsymbol{s}_2 := \boldsymbol{r}_2/\|\boldsymbol{r}_2\| \in \mathbb{R}^N$ under $\boldsymbol{r}_2 := \boldsymbol{R}\boldsymbol{p} - \langle \boldsymbol{R}\boldsymbol{p}, \boldsymbol{s}_1\rangle \boldsymbol{s}_1 \neq \boldsymbol{0}$.

Proposition 2 *Suppose that* $\boldsymbol{p}(\neq \boldsymbol{0})$ *is not an eigenvector of* $\boldsymbol{R}(\neq \boldsymbol{O})$ *(*$\Leftrightarrow \boldsymbol{r}_2 \neq \boldsymbol{0}$*). Then,*

$$\sqrt{\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_2\})} = \frac{\sqrt{\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{Rp}\})} - \sqrt{\mathrm{En}(\boldsymbol{Rp}, \mathcal{R}\{\boldsymbol{s}_1\})\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\})}}{\sqrt{1 - \mathrm{En}(\boldsymbol{Rp}, \mathcal{R}\{\boldsymbol{s}_1\})}}.$$

Referring to Proposition 2, the loss of energy could partly be compensated in $\mathcal{R}\{\boldsymbol{s}_2\}$ because a small value of $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_1\})$ increases the value of $\mathrm{En}(\boldsymbol{h}_{\mathrm{MMSE}}, \mathcal{R}\{\boldsymbol{s}_2\})$. However, it is not expected that the energy loss is fairly compensated because the large eigenvalue-spread also affects the other energy terms. In our experiments, we observe that, for strongly correlated input signals, $\bar{\boldsymbol{h}}^*$ tends not to have a sparse structure. In such a case, the whitening process is greatly helpful.

3. NUMERICAL EXAMPLES

This section provides simulation results to show the efficacy of the proposed scheme. Since Scheme 1 and its simplified version (see Sec. 2.3) exhibit comparable performance, we employ the simplified scheme. $U = 300$ independent simulations are performed for $N = 50$ with a random estimandum and white input signals under SNR $:= 10\log_{10}\left(E\left\{z_k^2\right\}/E\left\{n_k^2\right\}\right) = 30$ dB, where $z_k := \boldsymbol{u}_k^T\boldsymbol{h}^*$ [see (1)]. The MSE and the system mismatch (i.e., normalized coefficient error) are computed as averages of the U simulations.

For all the algorithms employed, we set the step size $\lambda_k = 0.1$, $\forall k \in \mathbb{N}$. For the MWF-NLMS [8], we use $D = 4, 8, 12$, $\gamma = 0.99$, $\widehat{\boldsymbol{R}}_0 = 0.01\boldsymbol{I}$, and $\widehat{\boldsymbol{p}}_0 = \boldsymbol{0}$. We use the same parameters for the proposed scheme. Moreover, for the proposed scheme, we set $\eta = 0.5$, $\varepsilon = 0.0001$, $K_0 = 100$, and $\varsigma_k = 0.05$, $\forall k \in \mathbb{N}$.

The results are depicted in Fig. 2. We see that, although the MWF-NLMS algorithm shows the fastest initial learning speed among the methods tested, its steady-state performance is far from the optimal. This is because MWF-NLMS seeks for a best solution in the lower dimensional subspace. On the other hand, the proposed scheme achieves reasonably fast initial learning speed and the optimal steady-state performance at the same time.

4. CONCLUSION

This paper has proposed *the Krylov-proportionate normalized least mean-square (KPNLMS) algorithm* attaining fast convergence with $O(N)$ complexity for an arbitrary estimandum. Also the paper has studied the energy of the MMSE solution in the Krylov subspace; the energy in a small dimensional Krylov subspace is closely connected to the performance of the proposed scheme. The numerical examples have demonstrated the significant advantages of the proposed scheme over the existing algorithms.

Acknowledgment
The author would like to thank warmly Prof. I. Yamada of Tokyo Institute of Technology for fruitful discussions.

5. REFERENCES

[1] J. S. Goldstein, I. S. Reed, and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Trans. Signal Processing*, vol. 44, no. 7, pp. 2943–2959, Nov. 1998.

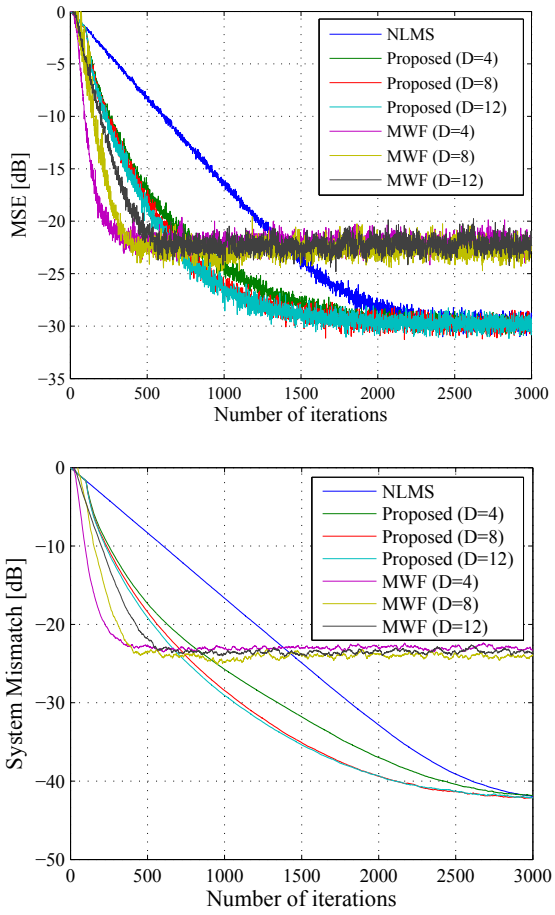


Fig. 2. Proposed versus MWF-NLMS and the standard NLMS algorithm under SNR 30 dB.

[2] M. L. Honig and W. Xiao, "Performance of reduced-rank linear interference suppression," *IEEE Trans. Inform. Theory*, vol. 47, no. 5, pp. 1928–1946, July 2001.

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

[4] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE ICASSP*, 2002, pp. 1881–1884.

[5] 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, July 2007.

[6] M. Yukawa and I. Yamada, "Adaptive parallel variable-metric projection algorithm —An application to acoustic echo cancellation," in *Proc. IEEE ICASSP*, 2007, pp. 1353–1356.

[7] 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.

[8] M. L. Honig and J. S. Goldstein, "Adaptive reduced-rank interference suppression based on multistage Wiener filter," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 986–994, Jun. 2002.