A Simplified Normalized Subband Adaptive Filter (NSAF) with NLMS-like complexity

John Håkon Husøy
Department of Electrical Engineering and Computer Science
University of Stavanger, Stavanger, Norway
john.h.husoy@uis.no

2022 International Conference on Applied Electronics (AE) | 978-1-6654-9482-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/AE54730.2022.9919894

***Abstract*—The Normalized Subband Adaptive Filter (NSAF) is a popular algorithm exhibiting moderate computational complexity and enhanced convergence speed relative to the ubiquitous Normalized Least Mean Square (NLMS) algorithm. Traditionally, the NSAF has made use of sophisticated perfect reconstruction (PR) filter banks and a block updating scheme, in which the adaptive filter vector is updated once every N samples, with N being equal to the number of subbands. Here we argue, first from a theoretical point of view, that an extremely simple two band filter bank with the simplest possible length 2 FIR filters, $\{1,-1\}$ and $\{1,1\}$, can be successfully used either with a sample by sample adaptive filter update, or with a block update performed for every second input signal sample. We demonstrate that this scheme actually works well through simulations. In short we obtain better convergence performance than the NLMS with a (multiplicative) computationally complexity proportional to $2M$, M being the length of the adaptive filter to be identified, with the block update and even better performance if we are willing to accept a computational complexity proportional to $4M$.**

Keywords*—*adaptive filter, subband adaptive filter, NLMS, NSAF

I. INTRODUCTION

The main goals in adaptive filter research over the years have been the conception of algorithms exhibiting *fast convergence* to the solution the the *Wiener-Hopf (W-H)* equation based on signals being realizations of wide sense stationary stochastic processes while minimizing the algorithms' *computational complexities*. The usefulness of adaptive filters in general as well as surveys of the most important algorithms and their properties are well documented and available in standard textbooks on the subject, [1], [2] and [3] being representative examples.

Unfortunately, the price for better convergence properties is the increase in computational complexity. The simplest algorithms – in conceptual, implementational terms as well as in terms of computational requirements – are the *Least Mean Square* (LMS) algorithm and its normalized version, the NLMS algorithm. They are characterized by a computational complexity proportional to $2M$, M being the length of the adaptive filter to be identified when using the algorithm in a system identification setting. The price for this simplicity is poor convergence speed when the input signal is highly colored. Many papers on improved adaptive filtering algorithms demonstrate their superior performance relative to the (N)LMS by using first order autoregressive (AR(1)) signals with high sample to sample correlation as examples. What we aim to do in this paper is to devise an improved NLMS-like algorithm making use of simple ideas based on the *Normalized Subband Adaptive Filter* (NSAF) [4] and making use of the important insights of [5], [6] in which for the first time we presented a quantitative tool for predicting the NSAF's convergence speed. Based on this we propose what we have called the simplified NSAF algorithm with two subbands using extremely simple filter banks and having a (multiplicative) computational complexity proportional to either $4M$ or $2M$.

We have organized our paper as follows: In the next section we present some necessary background on how to interpret adaptive filters as a Richardson iteration [7] applied to the W-H equation using instantaneous estimates for statistical quantities. Subsequently we show how the NSAF fits into this framework. This material derives partially from [5], [6], [8]. Having observed that the NSAF's convergence is determined by a preconditioning matrix formed as a linear combination of Toeplitz matrices, we put forward the idea, in the following section, that we can design *single* optimal Toeplitz preconditioning matrices for signals with given autocorrelation properties. Following this, we present our proposed algorithm and argue for its superior performance when compared the NLMS. This is followed by a section in which we demonstrate the algorithm's salient properties. Finally, we summarize and conclude our paper. **Note: It is possible to appreciate the results of this paper without reading all the details of the theoretical justification for the algorithm. Thus, the main content is a precise justification for the use of extremely simple filter banks based on the two $2-$tap FIR filters $\{1,-1\}$ and $\{1,1\}$, the simplest of all possible high-/low-pass filters. In other words, if one accepts the NSAF as such, we point out that the simplified version based on (17) and (18) works quite well. That the complexity is as pointed out above is almost self evident from these equations.**

II. THE WIENER-HOPF EQUATION, RICHARDSON ITERATIONS AND ADAPTIVE FILTERS

In this paper, we shall find it convenient to deal with the adaptive filtering problem in a system identification context, i.e., we shall assume that the input signal to our filter, $x(n)$, gives rise to a so-called *desired* signal through the linear model

$$d(n) = \underline{x}^T(n) \cdot \underline{h}_t + v(n), \tag{1}$$

where $\underline{x}(n) = [x(n), x(n-1), \ldots, x(n-M+1)]^T$ is a length M column vector (all vectors are assumed to be column vectors unless explicitly transposed) of input signal samples[1] corresponding to signal $x(n)$ at various time instants, and $\underline{h}_t$ is the length M vector representing what we shall refer to as the true Wiener filter (to be found by our adaptive filter algorithm). $v(n)$ is additive noise which is assumed to be uncorrelated with $x(n)$.

If $\mathbf{Q}^T$ is a full rank $M \times \{M + 2(L_f - 1)\}$ matrix – the significance of integer parameter L_f will be explained later – we see, upon multiplying both sides of (1) with $\mathbf{Q}^T \bar{\bar{\underline{x}}}(n)$, with $\bar{\bar{\underline{x}}}(n) = [x(n+L_f-1), x(n+L_f-1), \ldots, x(n), x(n-1), \ldots, x(n-M-1), \ldots, x(n-M-L_f+2]^T$ being an $\{M + 2(L_f - 1)\}$-vector, taking expectations on both sides of the equality sign and making use of the uncorrelatedness of $x(n)$ and $v(n)$, that $\underline{h}_t$ satisfies what we will refer to as the *preconditioned, augmented* Wiener-Hopf equation

$$\mathbf{Q}^T \bar{\bar{\mathbf{R}}} \underline{h}_t = \mathbf{Q}^T \bar{\bar{\underline{r}}}_{xd}, \tag{2}$$

where $\bar{\bar{\mathbf{R}}}$ is an $\{M + 2(L_f - 1)\} \times M$ rectangular autocorrelation

[1]Warning on notation: As is common, we do not use any notation to distinguish the cases when $\underline{x}(n)$ is to be interpreted as a random vector and when it is to be interpreted as a vector of signal samples.

978-1-6654-9482-3/22/$31.00 © 2022 IEEE

matrix of form

$$\bar{\bar{\mathbf{R}}} = \begin{bmatrix} r_{L_f-1} & r_{Lf} & \cdots & \cdots & r_{M+L_f-2} \\ r_{L_f-2} & r_{Lf-1} & \cdots & \cdots & r_{M+L_f-3} \\ \vdots & \vdots & & & \vdots \\ r_1 & r_2 & \cdots & \cdots & r_M \\ r_0 & r_1 & \cdots & \cdots & r_{M-1} \\ r_1 & r_0 & r_1 & \cdots & r_{M-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & r_1 \\ r_{M-1} & r_{M-2} & \cdots & \cdots & r_0 \\ r_M & r_{M-1} & \cdots & \cdots & r_1 \\ \vdots & \vdots & & & \vdots \\ r_{M+L_f-2} & r_{M+L_f-3} & \cdots & \cdots & r_{L_f-1} \end{bmatrix}, \quad (3)$$

with $r_{|m|} = E\{x(n)x(n+m)\}$ and

$$\bar{\bar{\underline{r}}}_{xd} = [\underline{r}_{xd}(-L_f+1), \ldots, \underline{r}_{xd}(-1), \underline{r}_{xd}(0), \underline{r}_{xd}(1), \ldots, \underline{r}_{xd}(M-1), \ldots, \underline{r}_{xd}(M+L_f-2)]^T, \quad (4)$$

with $\underline{r}_{xd}(m) = E\{x(n)d(n+m)\}$. Note that, if we set integer parameter $L_f = 1$ and $\mathbf{Q}^T = \mathbf{I}$, $\mathbf{I}$ being the $M \times M$ identity matrix, we get the 'ordinary' W-H equation familiar from any course on statistical signal processing, i.e.

$$\mathbf{R}\underline{h} = \underline{r}_{xd}, \quad (5)$$

where $\mathbf{R}$ is the autocorrelation matrix of the filter input signal, $\mathbf{R} = E\{\underline{x}(n)\underline{x}^T(n)\}$, and $\underline{r}_{xd}$ is the cross correlation vector defined by $\underline{r}_{xd} = E\{\underline{x}(n)d(n)\}$. We note in passing that $\bar{\bar{\mathbf{R}}}$ embeds the 'ordinary' $M \times M$ autocorrelation matrix $\mathbf{R}$ in its middle portion when $L_f > 1$. We stress that the solutions to (2) and (5) are identical.

We have previously argued that a conceptually simple way of deriving the LMS algorithm is to first apply a Richardson iteration to (5), i.e.[2]

$$\underline{h}(k+1) = \underline{h}(k) + \mu[\underline{r}_{xd} - \mathbf{R}\underline{h}(k)], \quad (6)$$

with μ being referred to as the step size. Subsequently, the statistical quantities $\mathbf{R}$ and $\underline{r}_{xd}$ are replaced by their instantaneous estimates at time k, $\underline{x}(k)\underline{x}^T(k)$ and $\underline{x}(k)d(k)$, respectively. With this we have

$$\underline{h}(k+1) = \underline{h}(k) + \mu\underline{x}(k)[d(k) - \underline{x}^T(k)\underline{h}(k)], \quad (7)$$

which is recognized as the classical LMS algorithm. Note that we can also argue in the opposite direction: Based on (7) we can surmise that $\underline{x}(k)\underline{x}^T(k)$ and $\underline{x}(k)d(k)$ are estimates of the true statistical quantities $\mathbf{R}$ and $\underline{r}_{xd}$ and that the convergence properties of the LMS algorithm (7) can be deduced from those of (6). This is essentially what is done in standard textbooks – although presented quite differently from what is done here – when analyzing the *convergence in the mean*[3] of the LMS leading to the conclusion that a low eigenvalue spread of $\mathbf{R}$, the eigenvalue spread being defined as the ratio $\lambda_{max}/\lambda_{min}$, gives rapid convergence, whereas a high eigenvalue spread for $\mathbf{R}$ implies slow convergence.

Applying a Richardson iteration to (2) yields

$$\underline{h}(k+1) = \underline{h}(k) + \mu\mathbf{Q}^T[\bar{\bar{\underline{r}}}_{xd} - \bar{\bar{\mathbf{R}}}\underline{h}(k)]. \quad (8)$$

We will now point out that the NSAF algorithm that is traditionally based on sophisticated perfect reconstruction (PR) filter banks [9] corresponds to (8) when instantaneous estimates are replaced by underlying true statistical correlation quantities. Thus, we surmise that the convergence speed of NSAF is dictated by the eigenvalue spread of $\mathbf{Q}^T\bar{\bar{\mathbf{R}}}$. This will then form the basis for our simple algorithm based on two simple filters with unit pulse responses of length 2: $\{1, -1\}$ and $\{1, 1\}$, i.e. the simplest possible low-/high-pass filters.

[2]We will prefer k as the iteration index, consequently it will be convenient also to relate this index to time.

[3]Which also involves invoking the so called independence hypothesis [3].

III. CONVERGENCE OF NSAF

The NSAF algorithm can be stated as follows [4]:

$$\underline{h}(k+1) = \underline{h}(k) + \mu\sum_{i=0}^{L-1}\frac{\underline{x}_i(k)e_{i,D}(k)}{\|\underline{x}_i(k)\|^2}, \quad (9)$$

where L is the number of subbands in the employed analysis filter bank characterized by L subband filters whose unit pulse responses are collected in length L_f column vectors $\underline{f}_i$ for $i = 0, 1 \ldots, L-1$. $\underline{x}_i(k)$ is the (*non-subsampled*) vector of M consecutive samples of input signal $x(n)$ filtered through $\underline{f}_i$. This quantity is computed each time N new input signal samples are available[4], i.e.,

$$\underline{x}_i(k) = \mathbf{F}_i^T\bar{\underline{x}}(kN), \quad (10)$$

where the $M \times \{M + L_f - 1\}$ filtering matrix has Toeplitz form:

$$\mathbf{F}_i^T = \begin{bmatrix} \underline{f}_i^T & 0 & 0 & \cdots & 0 \\ 0 & \underline{f}_i^T & 0 & \cdots & \\ \vdots & \ddots & \underline{f}_i^T & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \underline{f}_i^T \end{bmatrix}, \quad (11)$$

and where $\bar{\underline{x}}(kN)$ is an input signal vector of length $M + L_f - 1$ with first element $x(kN)$. With $\bar{\underline{d}}^T(n) = [d(n), d(n-1), \ldots, d(n - L_f + 1)]$, the other entities of (9) can be expressed in linear algebra terms as follows:

$$e_{i,D}(k) = [\bar{\underline{d}}^T(kN)\underline{f}_i - \bar{\underline{x}}^T(kN)\mathbf{F}_i\underline{h}(k)], \quad (12)$$

and

$$\begin{aligned} \|\underline{x}_i(k)\|^2 &= \underline{x}_i^T(k)\underline{x}_i(k) = \bar{\underline{x}}^T(kN)\mathbf{F}_i\mathbf{F}_i^T\bar{\underline{x}}(kN) \\ &= \sum_{m=0}^{M-1}\underline{f}_i^T\bar{\underline{x}}(kN-m)\bar{\underline{x}}^T(kN-m)\underline{f}_i, \end{aligned} \quad (13)$$

where $\bar{\underline{x}}(n)$ is a signal vector of length L_f with $x(n)$ as its first element. This signal vector's $L_f \times L_f$ associated autocorrelation matrix to be used below is denoted $\bar{\mathbf{R}}$.

Based on the above we have shown in [5] that, when appropriately identifying various quantities as instantaneous estimates of underlying statistical quantities, the NSAF (9) stands in the same relation to (8) as the LMS algorithm (7) to the Richardson iteration of (6) when $\mathbf{Q}^T$ is as given below. This means that the convergence speed can be predicted quantitatively through the eigenvalue spread of $\mathbf{Q}^T\bar{\bar{\mathbf{R}}}$ and that, when comparing the expected convergence speed to that of the NLMS algorithm, it is appropriate to compare this eigenvalue spread to that of $\mathbf{R}$.

The specific form of $\mathbf{Q}^T$ for the NSAF is shown to be [5]

$$\mathbf{Q}^T = \sum_{i=0}^{L-1}\mathbf{Q}_i^T = \sum_{i=0}^{L-1}\|\underline{f}_i\|_{\bar{\mathbf{R}}}^{-2}\mathbf{F}_i^T\bar{\mathbf{F}}_i{}^{(R)^T}, \quad (14)$$

where, still according to [5], $\bar{\mathbf{F}}_i{}^{(R)^T}$ is an $\{M + L_f - 1\} \times \{M + 2(L_f - 1)\}$ matrix having the same structure as $\mathbf{F}_i^T$ except that the rows are the unit pulse responses $\underline{f}_i^T$ *flipped in left/right direction*, i.e., $\underline{f}_i^T \cdot \mathbf{J}$, where $\mathbf{J}$, the exchange matrix, is the matrix with 1's along the antidiagonal and 0's otherwise. Given the Toeplitz structure of $\mathbf{F}_i^T$ and $\bar{\mathbf{F}}_i{}^{(R)^T}$ and the length, L_f, of the non-zeros parts of each row, both $\mathbf{Q}_i^T$, $\mathbf{Q}^T$ and $\mathbf{F}_i^T\bar{\mathbf{F}}_i{}^{(R)^T}$ will be Toeplitz . The non-zero elements of the rows of $\mathbf{F}_i^T\bar{\mathbf{F}}_i{}^{(R)^T}$ are given by the correlation of the unit pulse response $\underline{f}_i$ with itself. There will consequently be $2L_f - 1$ non-zero

[4]There is an important distinction to the traditional presentation of NSAF implicit here in that we decouple the number of subbands, L from the 'subsampling' factor N which we would more appropriately call the block size of the algorithm. In the traditional NSAF literature we always have the situation $N = L$.

elements resulting from this correlation. The final unidentified quantity in (14) is

$$\|\underline{f}_i\|^2_{\bar{\mathbf{R}}} = \underline{f}_i^T \bar{\mathbf{R}} \underline{f}_i, \tag{15}$$

where we have used the definition of the $\bar{\mathbf{R}}$-weighted norm of a vector. Another finding of [5] was that the PR restriction on the filter banks was not necessary.

IV. OUR ALGORITHM PROPOSAL AND ITS JUSTIFICATION

A good preconditioner $\mathbf{Q}^T$ in the context of (2) and (8) is such that $\mathbf{Q}^T\tilde{\bar{\mathbf{R}}} \approx \mathbf{I}$. One way of obtaining such a good preconditioner for an input signal with specific autocorrelation properties encapsulated in a specific $\tilde{\bar{\mathbf{R}}}$, is to seek the $\mathbf{Q}^T$ matrix minimizing

$$||\mathbf{Q}^T\tilde{\bar{\mathbf{R}}} - \mathbf{I}||_F^2, \tag{16}$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm.

Restricting the $\mathbf{Q}^T$ matrix to be Toeplitz – which seems reasonable in view of what was stated at the end of the previous section – it turns out that for low to moderate order autoregressive (AR) signals we can find extremely simple preconditioners giving perfect preconditioning. By way of example, consider the AR(1) signal $x(n) = \rho x(n-1) + w(n)$, where $w(n)$ is white and gaussian with $E\{w^2(n)\}$ selected such that $E\{x^2(n)\} = \sigma_x^2 = 1$. Solving the minimization problem in (16) with[5] $\rho = 0.9$, we find that the first row of $\mathbf{Q}^T$ (keep its Toeplitz structure in mind) is given by $[-4.7368, 9.5263, -4.7368]$ followed by the appropriate number of zeros. We have found – through experiments in Matlab – that this result is independent of L_f as long as it is specified to be larger than 1 and that it is also independent of M. Thus, we fix $L_f = 2$ in the following, keeping in mind that this means that the length of the non-zero parts of the rows of the $\mathbf{Q}^T$ and $\mathbf{Q}_i^T$ matrices are $2L_f - 1 = 3$. Also note that any scaled version of this vector, for example $[-0.9945, 2, -0.9945]$ will do the job that matters, namely reducing the ratio of the largest to the smallest eigenvalue to 1, i.e. we have perfect preconditioning with all eigenvalues being equal! Repeating what was done above, but now with $\rho = -0.9$, we find that the vector $[0.9945, 2, 0.9945]$ as the basis for the $\mathbf{Q}^T$ matrix also provides perfect preconditioning. With $M = 32$ the original eigenvalue spread of $\mathbf{R}$ with $\rho = \pm 0.9$ is 263.1 Of course, doing the iteration in (8) with an eigenvalue spread of 1 for the coefficient matrix $\mathbf{Q}^T\tilde{\bar{\mathbf{R}}}$ rather than the iteration in (6) with an eigenvalue spread of 263.1 for $\mathbf{R}$ gives dramatic improvements in convergence speed. It is well known that when selecting a preconditioner, there is always a tradeoff between computational complexity and performance. Although the above preconditioners are simple, if we choose them even simpler by approximation, i.e. we base the $\mathbf{Q}^T$ matrices instead on the vectors $[-1, 2, -1]$ and $[1, 2, 1]$ for $\rho = 0.9$ and $\rho = -0.9$, respectively, we find that the eigenvalue spread of $\mathbf{Q}^T\tilde{\bar{\mathbf{R}}}$, still with $M = 32$, is equal to 3.7. As we see, there is still a dramatic reduction in the eigenvalue spread relative to the original spread of 263.1 with an attendant improvement in convergence speed.

Of course, having *only one* such optimized or approximately optimized preconditioner works very well when the autocorrelation is known beforehand. This is mostly not the case and we will typically use multiple or as, implied by (14), a linear combination of preconditioners. So, our proposal is to base our NSAF-related iteration (8) with two $\mathbf{Q}_i^T$'s, one based on $[-1, 2, -1]$ and the other based on $[1, 2, 1]$.

Again, looking back at the structure of $\mathbf{Q}^T$ as given in (14) and looking at the $\mathbf{F}_i^T \bar{\mathbf{F}}_i{}^{(R)^T}$ part, it is evident, as is pointed out at the end of the previous section, that the non-zero part of each row of this matrix product is the correlation of the unit pulse response of $\underline{f}_i$ with itself. With length two unit pulse responses $\underline{f}_0 = [1, -1]$ and $\underline{f}_1 = [1, 1]$ it is evident that the correlation of each of these unit pulse responses with itself is given by $[1, 2, 1]$ and $[-1, 2, -1]$, respectively. Thus, the NSAF algorithm directly associated with these choices can be stated explicitly as

$$\begin{aligned}\underline{h}(k+1) &= \underline{h}(k) \\ &+ \mu \sum_{i=0}^{1} \frac{\mathbf{F}_i^T \tilde{\underline{x}}(kN)}{\|\mathbf{F}_i^T \tilde{\underline{x}}(kN)\|^2} \cdot [\bar{\underline{d}}^T(kN)\underline{f}_i - \tilde{\underline{x}}^T(kN)\mathbf{F}_i \underline{h}(k)],\end{aligned} \tag{17}$$

where, if we follow the NSAF literature, we have to set $N = 2$ (since we have two subbands). We point out here, however, that there is nothing preventing us from also selecting $N = 1$ in which case we have a sample-by-sample algorithm just as the NLMS. The signal vectors involved are defined previously, whereas the $\mathbf{F}_i^T$ matrices (of dimension $M \times \{M+1\}$ since we have set $L_f = 2$) for $i \in \{0, 1\}$ are

$$\mathbf{F}_i^T = \begin{bmatrix} 1 & \pm 1 & 0 & \cdots & 0 \\ 0 & 1 & \pm 1 & \cdots & \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 1 & \pm 1 \end{bmatrix}, \tag{18}$$

where the minus sign is used for $i = 0$ and the plus sign is used for $i = 1$. Given the above, and consequently that the use of these extremely simple filters does not require any multiplications, it is evident that the multiplicative complexity (for each new input sample) when $N = 1$ is proportional to $4M$ and proportional to $2M$ if $N = 2$.

Given the proposed algorithm we now proceed, in a semi-intuitive way, to explain why we should expect improved convergence speed relative to the NLMS. For this purpose we examine what happens when the input signal is 1) white, 2) lowpass, exemplified by an AR(1) signal with $\rho = 0.9$ and 3) highpass, exemplified by an AR(1) signal with $\rho = -0.9$.

1) If $\rho = 0$, i.e. we have a white input signal, we would like to have no preconditioning at all[6], since $\mathbf{R}$ is already diagonal with equal entries on its diagonal. Now, the $\|\underline{f}_i\|^2_{\bar{\mathbf{R}}} = \underline{f}_i^T \bar{\mathbf{R}} \underline{f}_i$ appearing as a denominator in (14) will be equal for $i = 0$ and $i = 1$ since $\bar{\mathbf{R}}$ is diagonal. Thus, according to (14) and with $\mathbf{Q}_0^T$ and $\mathbf{Q}_1^T$ based on vectors $[-1, 2, -1]$ and $[1, 2, 1]$, the preconditioner $\mathbf{Q}^T$ will be diagonal, which corresponds to our desire of not having any preconditioning at all. Therefore, we expect the algorithm to work just like the NLMS.
2) If $\rho = 0.9$, i.e. we have a lowpass signal, we would ideally – see the discussion above – like to have a single preconditioner $\mathbf{Q}^T$ based on vector $[-1, 2, -1]$. Since (14) implies a linear combination of two preconditioners, this cannot be achieved. However since for lowpass signals $\|\underline{f}_0\|^2_{\bar{\mathbf{R}}} < \|\underline{f}_1\|^2_{\bar{\mathbf{R}}}$ the resulting preconditioner will be dominated by $\mathbf{Q}_0^T$ based on the desired vector, $[-1, 2, -1]$. In this case the resulting preconditioner, still with $M = 32$, results in $\mathbf{Q}^T\tilde{\bar{\mathbf{R}}}$ having an eigenvalue spread of 14.14, which of course is worse than the 3.7 reported above for a single preconditioner, but still constitutes a dramatic improvement when comparing to the original eigenvalue spread of 263.1.
3) If $\rho = -0.9$, i.e. we have a highpass signal. Based on the same argument as above, but with $\|\underline{f}_0\|^2_{\bar{\mathbf{R}}} > \|\underline{f}_1\|^2_{\bar{\mathbf{R}}}$, the resulting preconditioner will, as desired, be dominated by $\mathbf{Q}_1^T$ based on the vector $[1, 2, 1]$. As in the case above we get an eigenvalue spread of 14.14.

Based on the above, we would expect significant convergence improvements relative to the NLMS for some signal characteristics, and hopefully some improvements for most other signal characteristics.

V. EXPERIMENTS AND EVALUATION

In this section we present a limited number of illustrative simulation results in a system identification context where the unknown system

[5]Common choice for ρ used in the literature when the objective is to demonstre the superior performance of some adaptive algorithm relative to that of the NLMS.

[6]The eigenvalue spread is already minimal, it cannot be further reduced.

(the Wiener solution, $\underline{h}_t$) is assumed to be an FIR filter of length $M = 32$. We have two designed versions of the 'true' filter, first we specify $\underline{h}_t$ as a lowpass filter, then as a highpass filter. We obtain these filters through the Matlab commands[7] fir1(30,0.4) and fir1(30,0.4,'high'), respectively. In the implementation of (17), as well as in the implementation of the NLMS used for comparision, we set $\mu = 0.5$, the signal $d(n)$ of (1) is generated with $E\{v^2(n)\} = \sigma_v^2 = 10^{-4}$. As performance metric we employ the normalized Mean Square Deviation (MSD) defined as

$$\text{MSD}(k) = \frac{\|\underline{h}(k) - \underline{h}_t\|^2}{\|\underline{h}_t\|^2}, \tag{19}$$

averaged over 100 runs of the respective algorithms.

We have considered several input signal models:

1. White signal; $x(n) = w(n)$.
2. lowpass signal conforming to $x(n) = 0.9x(n-1) + w(n)$.
3. highpass signal conforming to $x(n) = -0.9x(n-1) + w(n)$.
4. Signal conforming to $x(n) = 0.1x(n-1) + 0.8x(n-2) + w(n)$, which has significant lowpass as well as highpass content.
5. Signal conforming to $x(n) = 1.6x(n-1) - 0.81x(n-2) + w(n)$, which is lowpass with a significant peak around $\omega = 0.5$.

As a first indication of convergence speed improvements to be expected, we have computed the theoretical eigenvalue spread of $\mathbf{R}$ and $\mathbf{Q}^T\tilde{\mathbf{R}}$ associated with the simplified NSAF. The results are shown in Table I. As hoped for, we get moderate to significant reductions in eigenvalue spread which should manifest itself in increased convergence speed. We have also done this for some more complicated AR(10) and AR(16) signals in which case we get reductions in eigenvalue spread of about a factor of 10.

Signal model	Orig. $\lambda_{max}/\lambda_{min}$	Prec. $\lambda_{max}/\lambda_{min}$
No. 1	1.0	1.0
No. 2	263.1	14.14
No. 3	263.1	14.14
No. 4	173.76	93.08
No. 5	1160.90	128.78

TABLE I
EIGENVALUE SPREAD FOR $\mathbf{R}$ INDICATIVE OF THE CONVERGENCE SPEED FOR NLMS AND EIGENVALUE SPREAD FOR $\mathbf{Q}^T\tilde{\mathbf{R}}$ INDICATIVE OF THE CONVERGENCE SPEED FOR SIMPLIFIED NSAF.

We now proceed to show some plots of the MSD(k) to see if the reductions in eigenvalue spreads manifest themselves in improved convergence speeds relative to the NLMS. In Figure 1 – smpl in the figure captions refer to the simplified NSAF, i.e. (17) and (18) – we have shown results for model 2 with $\underline{h}_t$ being lowpass. As we see, we have significant improvements in convergence speed relative to the NLMS. We observed similar behavior with the model 3 and $\underline{h}_t$ being either low- or highpass. Somewhat more modest performance improvements were observed when $\underline{h}_t$ is highpass with model 2 or $\underline{h}_t$ is lowpass with model 3. For all the models considered we observed moderate to good performance improvements relative to the NLMS, even for $N = 2$. The least performance improvement we observed was for model 4 with $\underline{h}_t$ being highpass, see Figure 2 for the MSD(k) curves. As a final example, we show, in Figure 4 the MSD(k) curves for an AR(16) signal whose spectrum is shown in Figure 3. Once again, we observe distinctive improvements in the convergence speed. In a way, there seems to "be a free lunch" in the sense that, in the $N = 2$ case, we get mostly better convergence speed than the NLMS without paying a price in terms of increased computational complexity.

[7]A zero is appended at the end to get filter length $M = 32$.

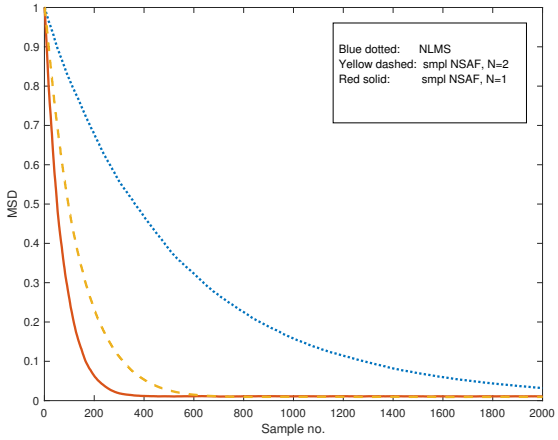

Fig. 1. MSD for model 2 with h_t being a lowpass filter.

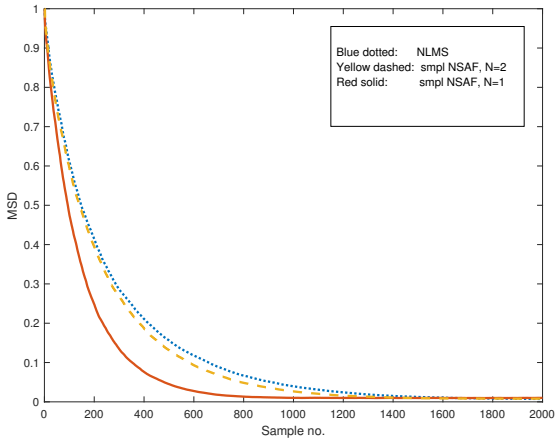

Fig. 2. MSD for model 4 with h_t being a highpass filter.

VI. CONCLUSION

From previous insights into the NSAF algorithm [5], [6], we have been able to suggest simplified NSAF algorithms with very low computational complexity. We have demonstrated that these algorithms present advantages in terms of convergence speed relative to the NLMS while retaining this algorithm's low computational complexity.

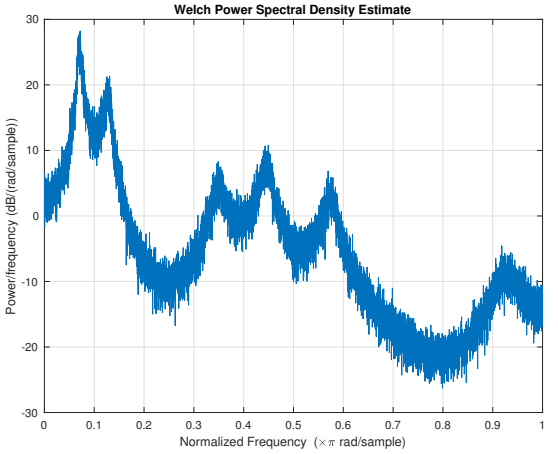

Fig. 3. Spectrum of AR(16) model with several poles very close to the unit circle.

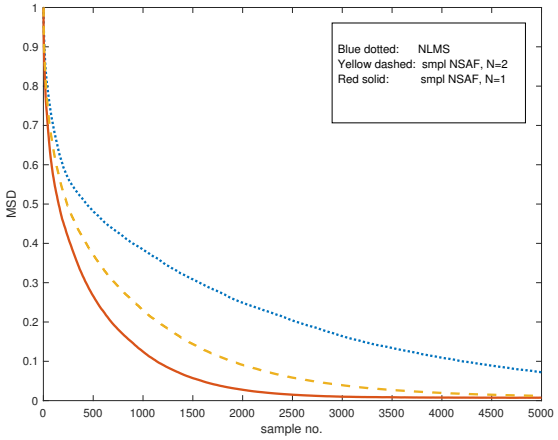

Fig. 4. MSD for AR(16) signal with spectrum as shown in Figure 3 with h_t being a highpass filter.

REFERENCES

[1] A. H. Sayed, *Adaptive Filters*. Wiley, 2008.

[2] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, 4th ed. Springer, 2013.

[3] S. Haykin, *Adaptive Filter Theory*, 4th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[4] K. A. Lee and W. S. Gan, "Improving convergence of the NLMS algorithm using constrained subband updates," *IEEE Signal Processing Letters*, vol. 11, pp. 736–739, Sep. 2004.

[5] J. H. Husøy and M. S. E. Abadi, "On the convergence speed of the normalized subband adaptive filter: Some new insights and interpretations," in *Proc. of Thirteenth Intl. Symp. on Signals, Circuits and Systems*, Iasi, Romania, Jul. 2017.

[6] J. H. Husøy, "On the selection and design of filter banks in normalised subband adaptive filters (NSAF)," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2017 Asia-Pacific*, Kuala Lumpur, Malaysia, Dec 2017, pp. 877–883.

[7] Y. Saad, *Iterative Methods for Sparse Linear Systems*. PWS Publishing, 1996.

[8] J. H. Husøy and M. S. E. Abadi, "A unified approach to adaptive filters and their performances," *IET Signal Processing*, vol. 2, pp. 97 – 109, Jun. 2008.

[9] K.-A. Lee, W.-S. Gan, and S. M. Kuo, *Subband Adaptive Filtering: Theory and Implementation*. Wiley Publishing, 2009.