Mean Square Behavior of Noise-Robust Normalized Subband Adaptive Filter Algorithm
Yi Yu1, Haiquan Zhao2, Badong Chen3, Wenyuan Wang3, Lu Lu4

1 School of Information Engineering, Southwest University of Science and Technology, Mianyang, China
2 School of Electrical Engineering, Southwest Jiaotong University, Chengdu, China
3 School of Electronic and Information Engineering, Xi’an Jiaotong University, Xi’an, China
4 School of Electronics and Information Engineering, Sichuan University, Chengdu, China

Abstract: This paper studies the statistical models of the noise-robust normalized subband adaptive filter (NR-NSAF) algorithm in the mean and mean square deviation senses involving transient and steady-state behavior by resorting to the vectorization and Kronecker product of matrices. Thus, the proposed analysis does not require the Gaussian assumption to the input signal. Moreover, it removes the paraunitary assumption aiming to the analysis filter banks as in the existing analyses of subband adaptive algorithms. Simulation results in various conditions demonstrate the effectiveness of our theoretical analysis. For a special form of the algorithm, the proposed steady-state expression is also better accurate than the previous analysis.

1. Introduction
Adaptive filter algorithms have a pivotal position in some applications such as system identification, channel equalization, active noise control, and echo cancellation [1], [2]. One of the popular algorithms is the normalized least mean square (NLMS), and it is simple and easy in implementation. Nevertheless, the problem is very slow convergence rate for the correlated input signal. To overcome this problem, subband adaptive filter (SAF) with multiband structure has attracted much attention due to its decorrelation property. The multiband structure eliminates aliasing and band edge effects relative to the conventional structure [3]. According to this, the normalized SAF (NSAF) algorithm was proposed by Lee and Gan in [3]. Compared with the NLMS, the NSAF has faster convergence rate when the input signal has high correlation in the time-domain, while retaining comparable computational complexity. In a recent decade, many works have been reported to further obtain an improvement on the performance of the NSAF [4]-[12]. Typically, inspired by the NLMS with reusing weight vectors at each update [13], a noise-robust NSAF (NR-NSAF) algorithm was proposed which improves the steady-state performance in highly noisy environments [8], and almost at the same time, Ni proposed an improved NSAF (INSAF) algorithm [9]. And, the NR-NSAF algorithm is a more general form of the INSAF algorithm.

The performance analysis is a crucial point in the study of adaptive filter algorithms [2], [14]-[18]. Much literature has addressed the performance analysis of the NSAF algorithm [19]-[23]. Specifically, the steady-state mean-square error (MSE) of the NSAF using a fixed step size and a fixed regularization parameter were studied in [19] and [20], respectively. In some applications, e.g., system identification and echo cancellation, adaptive filter estimates the impulse response of the underlying system, so studying the mean square deviation (MSD) performance of adaptive algorithms seems to be more appropriate than the MSE. In general, the MSE can also be obtained from the MSD through the autocovariance matrix of input vector. In [24], Jeong et al. analyzed the steady-state MSD of the INSAF algorithm and this analysis framework has also been extended to its under-modeling scenario [25] and affine projection variant [10]. The theoretical results coincides with the simulations, but the accuracy depends on large number of subbands and long adaptive filter. However, the transient behavior of the INSAF algorithm has not been studied.

In this paper, we analyze the MSD performance of the NR-NSAF algorithm. Our analysis is based on the method of the vectorization and Kronecker product of matrices developed originally by Sayed [2]. This method is very popular recently, since it does not enforce the input signal to following a specified model (e.g., Gaussian distribution). Our contributions are summarized as below: 1) analyzing the transient and steady-state MSD of the NR-NSAF algorithm; 2) providing the mean condition on the step-size to ensure the algorithm stability. Moreover, unlike the existing analyses of subband adaptive algorithms, the proposed analysis does not assume the analysis filter banks to being paraunitary. Extensive simulations verify the proposed theoretical results.

Notations: (·)T, ||·||2, λmax(·), E{·}, Tr(·), ρ(·), and ⊗ denote the transpose operator, the Euclidean norm of a vector, the largest eigenvalue of a matrix, the expectation of random variables, the trace of a matrix, the spectral radius of its matrix argument, and the Kronecker product, respectively. The notation diag{·} yields the diagonal matrix according to its vector argument. The vectorization operator vec(·) transforms an $M \times M$ matrix into an $M^2 \times 1$ column vector by stacking successively the columns of the matrix, and vec⁻¹(·) is the inverse of vec(·). The symbols 1 and 0 denote the identity and zero matrices with appropriate sizes, respectively. Also, all the vectors in this paper are column vectors.
2. The NR-NSAF Algorithm

Supposing that the desired signal \( d(n) \) is given by the linear model with respect to the input signal \( u(n) \):
\[
d(n) = u^T(n)w_o + \eta(n),
\]
where \( w_o \) is an unknown \( M \)-length vector that needs to be identified, \( u(n) = [u(n), u(n-1), ..., u(n-M+1)]^T \) is the input vector, and \( \eta(n) \) is the system noise. Fig. 1 shows the multiband-structured SAF with \( N \) subbands. The input signal \( u(n) \) and the desired signal \( d(n) \) are partitioned into multiple subband signals \( u_i(n) \) and \( d_i(n) \) by the analysis filters \( H_i(z) \), \( i = 0, 1, ..., N-1 \), respectively. The subband outputs \( y_i(n) \) are obtained by filtering signals \( u_i(n) \) through a fullband adaptive filter whose weight vector is denoted as \( w(k) = [w_1(k), w_2(k), ..., w_M(k)]^T \). Then, by \( N \)-fold decimating signals \( y_i(n) \) and \( d_i(n) \), we obtain \( y_{i,D}(n) \) and \( d_{i,D}(n) \) at lower sampling rate, respectively. We use \( n \) and \( k \) to represent the original sequences and the decimated sequences, respectively. For the \( i \)-th subband, the decimated error signal is expressed as
\[
e_{i,D} = d_{i,D}(n) - u_i^T(n)w_o(k),
\]
where \( u_i(n) = [u_i(n), u_i(n-1), ..., u_i(n-M+1)]^T \) and \( d_{i,D}(n) = d_i(n) \).

NR-NSAF algorithm reduces to the INSAF and NSAF algorithms when \( \alpha = 1 \) and \( P = 1 \), respectively.

3. Performance analysis

Let us introduce two matrices:
\[
U(k) = [u(kN), u(kN-1), ..., u(kN-L+1)]^T,
\]
\[
H = [h_0, h_1, ..., h_{N-1}]^T,
\]
and two vectors:
\[
d(k) = [d(kN), d(kN-1), ..., d(kN-L+1)]^T,
\]
\[
u(kN) = [u(kN), u(kN-1), ..., u(kN-M+1)]^T,
\]
where \( h_i \) is the impulse response of the \( i \)-th analysis filter \( H_i(z) \) with length of \( L \). Then, we can find the following relations:
\[
U_o(k) \triangleq u_0(k), u_1(k), ..., u_{N-1}(k) = U(k)H,
\]
\[
d_o(k) \triangleq [d_{0,D}(n), d_{1,D}(n), ..., d_{N-1,D}(n)]^T = H^Td(k),
\]
\[
d(k) = U^T(k)w_o + \eta(k),
\]
where \( \eta(k) = [\eta(kN), \eta(kN-1), ..., \eta(kN-L+1)]^T \). We use (4)-(6) to rearrange (3) as
\[
w(k+1) = \sum_{p=0}^{P-1} \beta_p w(k-p) + \mu U(k)HA^{-1}(k)
\]
\[
\begin{bmatrix}
H^T d(k) - H^T U^T(k) \sum_{p=0}^{P-1} \beta_p w(k-p)
\end{bmatrix},
\]
where \( A(k) = cI_n + \text{diag} \left[ U_o^T(k)U_o(k) \right] \).

Subtracting \( w_o \) from both sides of (8) yields
\[
\tilde{w}(k+1) = \sum_{p=0}^{P-1} \beta_p \tilde{w}(k-p) - \mu b(k),
\]
where \( \tilde{w}(k) \triangleq w_o - w(k) \) represents the weight error vector as,
\[
A(k) = U_o(k)A^{-1}(k)U_o^T(k) \quad \text{and} \quad b(k) = U_o(k)A^{-1}(k)H^T \eta(k).
\]
For deriving (9), we also use the relation \( \sum_{p=0}^{P-1} \beta_p = 1 \).

Before proceeding, we define some block matrices and vectors:
\[
\mathbf{A}(k) = \begin{bmatrix} A(k) & 0_{M \times (M-P)} \\ 0_{(M-P) \times M} & 0_{(M-P) \times (M-P)} \end{bmatrix},
\]
\[
\mathbf{B}(k) = \begin{bmatrix} b(k) \\ 0_{M \times (P-1)} \end{bmatrix}, \quad \text{where} \quad \beta = [\beta_0, ..., \beta_{P-1}] \otimes I_M.
\]

With these definitions, (9) can be rewritten as
\[
\tilde{W}(k+1) = (I_{MP} - \mu \mathbf{A}(k))\tilde{W}(k) - \mu \mathbf{B}(k).
\]
Equation (10) will be the starting point to analyze the performance of the NR-NSAF algorithm.

For tractable analysis, the following assumptions are necessary.

A1): The system noise \( \eta(n) \) is a white process with zero-mean and variance \( \sigma^2 \), which is independent of \( u(n) \).

A2): \( u(n) \) is zero-mean stationary random vector with
positive definite covariance matrix.

A3): \( \mathbf{u}_i(k) \) for \( i = 0, 1, \ldots, N - 1 \) and \( \mathbf{w}(k - p) \) for \( p = 0, \ldots, P - 1 \) are independent each other, which is the well-known independence assumption used for analyzing the performance of adaptive algorithms [2], [14]–[27].

From the assumption A2), \( \mathbf{u}_i(k) \) for \( i = 0, 1, \ldots, N - 1 \) are also zero-mean stationary with positive definite covariance matrices. According to assumptions A1) and A3), we can further assume that \( \mathbf{W}(k) \) is independent of \( \mathbf{A}(k) \) and \( \mathbf{B}(k) \).

### 3.1 Mean behavior

Under assumptions A1) and A3), the expectation of both sides of (10) is obtained:

\[
E\{ \mathbf{W}(k + 1) \} = (\mathbf{I}_{MP} - \mu E\{ \mathbf{A}(k) \}) \mathbf{CE} \{ \mathbf{W}(k) \}. \tag{11}
\]

**Theorem 1**: The NR-NSAF algorithm is mean stable if, and only if the step size satisfy

\[
0 < \mu < \frac{2}{\lambda_{\text{max}}(E\{ \mathbf{A}(k) \})}. \tag{12}
\]

**Proof**: See Appendix.

At the steady-state, i.e., when \( k \to \infty \), we obtain from (11):

\[
E\{ \mathbf{W}(\infty) \} = \mathbf{0}_{MP \times 1}. \tag{13}
\]

The above relation means that the NR-NSAF algorithm can yield an unbiased estimate for the unknown vector \( \mathbf{w}_o \).

### 3.2 Mean square behavior

Post-multiplying (10) with its transpose and defining \( \Phi(k) \triangleq \mathbf{W}(k)\mathbf{W}^T(k) \), the following matrix recursion is developed:

\[
\Phi(k + 1) = \mathbf{C} \Phi(k) \mathbf{C}^T - \mu \mathbf{A}(k) \mathbf{C} \Phi(k) \mathbf{C}^T
- \mu^2 \mathbf{A}(k) \mathbf{C} \Phi(k) \mathbf{C}^T \mathbf{A}(k)
+ \mu^2 \mathbf{B}(k) \mathbf{B}^T(k)
- \mu [\mathbf{I}_{MP} - \mu \mathbf{A}(k)] \mathbf{C} \mathbf{W}(k) \mathbf{B}^T(k)
- \mu \mathbf{B}(k) \mathbf{W}(k)^T \mathbf{C}^T [\mathbf{I}_{MP} - \mu \mathbf{A}(k)]^T. \tag{14}
\]

With assumptions A1-A3), the expectations of the last two terms in (14) are zero. Consequently, enforcing the expectation operator on both sides of (14), we have

\[
E\{ \Phi(k + 1) \} = \mathbf{CE} \{ \Phi(k) \} \mathbf{C}^T
- \mu E\{ \mathbf{A}(k) \} \mathbf{CE} \{ \Phi(k) \} \mathbf{C}^T
- \mu \mathbf{CE} \{ \Phi(k) \} \mathbf{C}^T E\{ \mathbf{A}(k)^T \}
+ \mu^2 \mathbf{E} \{ \mathbf{A}(k) \} \mathbf{CE} \{ \Phi(k) \} \mathbf{C}^T \mathbf{A}(k)
+ \mu^2 \mathbf{E} \{ \mathbf{B}(k) \} \mathbf{B}^T(k). \tag{15}
\]

The last expectation term of (15) can be further expressed as

\[
E\{ \mathbf{B}(k) \mathbf{B}^T(k) \} = \begin{bmatrix}
\sigma_w^2 \mathbf{I}_{M-P^2} & 0_{M \times (P-1)} \\
0_{(P-1) \times M} & 0_{M(P-1) \times (M(P-1))}
\end{bmatrix}, \tag{16}
\]

\[
E\{ \mathbf{b}(k)\mathbf{b}^T(k) \} = \sigma_w^2 \mathbf{I}_{M-P^2} \mathbf{U}_d(k) \mathbf{A}^H(k) \mathbf{H}^H \mathbf{H} \mathbf{A}^H(k) \mathbf{U}_d^H(k). \tag{17}
\]

\[
= \sigma_w^2 \sum_{i=0}^{N-1} |\mathbf{b}_i|^2 \mathbb{E}\left\{ \mathbf{U}_d(k) \mathbf{U}_d^H(k) + \sigma_w^2 \mathbf{I}_{M-P^2} \right\}.
\]

To continue the analysis, we need two properties of the Kronecker product, namely,

\[
\mathbf{vec}(\mathbf{X} \otimes \mathbf{Y}) = (\mathbf{Y}^T \otimes \mathbf{X}) \mathbf{vec}(\mathbf{Z}) \tag{18}
\]

and

\[
\mathbf{X} \otimes \mathbf{Z} = (\mathbf{X} \otimes \mathbf{Z}) (\mathbf{Y} \otimes \mathbf{Z}) \tag{19}
\]

for any matrices \( \{\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{\Omega}\} \) of compatible dimensions [28].

Subsequently, taking the vectorization for all the matrices in the recursion (15), after merging similar terms, it is established that

\[
\mathbf{vec} \{ E\{ \Phi(k + 1) \} \} = \mathbf{F} \mathbf{vec} \{ E\{ \Phi(k) \} \}
+ \mu^2 \mathbf{vec} \{ E\{ \mathbf{B}(k) \mathbf{B}^T(k) \} \}, \tag{20}
\]

where the \( M^2 P^2 \times M^2 P^2 \) matrix \( \mathbf{F} \) is given by

\[
\mathbf{F} = [\mathbf{I}_{MP} - \mu E\{ \mathbf{A}(k) \}] \mathbf{CE} \{ \mathbf{W}(k) \} \mathbf{B}^T(k)
+ \mu^2 \mathbf{CE} \{ \mathbf{A}(k) \} \mathbf{CE} \{ \mathbf{A}(k) \} \mathbf{C} \mathbf{C}^T. \tag{21}
\]

The MSD is defined as

\[
\operatorname{MSD}(k) \triangleq E\{ \mathbf{W}(k) \}^2 = \operatorname{Tr} E\{ \mathbf{W}(k) \mathbf{W}^T(k) \}
= \operatorname{Tr} \{ E\{ \Phi(k) \} \} \tag{22}
\]

where \( [ E \{ \Phi(k) \} ] \) is the first \( M \times M \) diagonal block of \( E \{ \Phi(k) \} \).

Thus, based on the inverse operator \( \mathbf{vec}^{-1}(\cdot) \), the recursion (20) models the MSD evolution behavior of the NR-NSAF algorithm with respect to the iteration \( k\).

It is seen from (21) that the NR-NSAF algorithm is convergent in the mean square sense if, and only if, the matrix \( \mathbf{F} \) is stable that all the eigenvalues of \( \mathbf{F} \) are in the range \((-1, 1)[14]\).

However, further obtaining the step size range from it is difficult due to the existence of the matrix \( (\mathbf{C} \otimes \mathbf{C}) \).

Fortunately, we have deduced the mean square convergence condition \( 0 < \mu < 2 \) in an alternative method, see Appendix in [12].

Then, if the algorithm converges to the steady-state, the equality \( E\{ \Phi(k + 1) \} = E\{ \Phi(k) \} \) as \( k \to \infty \) will be hold. Accordingly, we can arrive at the steady-state solution of (20):

\[
E\{ \Phi(\infty) \} = \mu^2 \mathbf{vec}^{-1}(\mathbf{I}_{MP} - \mathbf{F}) \mathbf{vec} \{ E\{ \mathbf{B}(k) \mathbf{B}^T(k) \} \}. \tag{23}
\]

Using the relation \( \operatorname{Tr}(\mathbf{XY}) = \mathbf{vec}(\mathbf{X}^T) \mathbf{vec}(\mathbf{Y}) \), the steady-state MSD of the NR-NSAF algorithm can be derived from (23), i.e.,

\[
\operatorname{MSD}(\infty) = \operatorname{Tr} E\{ \Phi(\infty) \} / P
= \mu^2 \mathbf{vec}^T(\mathbf{I}_{MP} - \mathbf{F}^{-1}) \times \operatorname{vec} E\{ \mathbf{B}(k) \mathbf{B}^T(k) \} / P. \tag{24}
\]

Remark 1. Reference [24] presents an MSD(\infty) expression.
for the simple form of the NR-NSAF algorithm when $\alpha=1$ and $\varepsilon=0$ (that is the INSAF algorithm). Nonetheless, it benefits from two extra assumptions: 1) when the number of subbands is sufficiently large, the decimated subband input signals are close to the white signals, i.e., $E[u_i(k)u_i^T(k)] \approx \sigma^2_{u_i} I_M$ and $E[u_i^T(k)u_i(k)] \approx M \sigma^2_{u_i}$; 2) The length $M$ of adaptive filter is large so that the fluctuation of the energy of the decimated subband input signals from one iteration to the next iteration is small. In addition, the MSD($\alpha$) in [24] requires knowing the variances of the subband noises, $\sigma^2_{u_i}$, which is usually given by $\sigma^2_{u_i} = \frac{\sigma^2_{u_i}}{N}$ under the paraunitary assumption for the analysis filters [21]. However, we propose to use $\sigma^2_{u_i} = ||h_i||^2 \sigma^2_{u_i}$.

4. Simulation results

Simulations are conducted in the system identification. Both the adaptive filter and the unknown vector have the same length $M=16$. The weight vector of adaptive filter is initialized as a null vector. The correlated input signal $u(n)$ is generated by filtering either a zero mean white Gaussian signal with unit variance or a uniform distribution signal with the interval $[-1, 1]$, through a first-order autoregressive system of a pole at 0.9 [22], called as the Gaussian input and the uniform input in simulations, respectively. The system noise $\eta(n)$ is a zero mean white Gaussian process, giving to a certain signal-to-noise ratio (SNR). The analysis filters are designed based on the cosine modulated filter banks, where the length of the prototype filter is 64 when the number of subbands is $N=8$, unless otherwise specified. For evaluating the proposed theoretical expressions, the expectations on the subband inputs shown in (17) and (21) are estimated by ensemble averaging. The regularization constant $\varepsilon$ is set to 0.001 except Fig. 8. All the simulations results are averaging over 200 independent trials.

4.1 Transient performance

To begin with, the mean evolution behavior of the algorithm is checked in Fig. 2 for identifying the unknown vector $w_o=[0.51, -0.04, 0.02, 0.09, 0.22, 0.20, 0.13, -0.48, -0.39, 0.32, -0.11, -0.30, 0.25, -0.24, 0.6, -0.01]^T$. It is clear to see that the theoretical weights calculated by (11) match well with the simulated weights.

In the following examples, we investigate the MSD performance of the algorithm by $10\log_{10} MSD(k)$ (dB). The unknown vector is randomly generated by using the function $rand(M,1)-0.5$ in MATLAB and normalized by $w_o^T w_o = 1$. Fig. 3 shows the effect of the parameter $\alpha$ on the NR-NSAF performance. As can be seen, the theoretical MSD curves computed by (20) have good agreement with the simulated curves. In addition, for values of $\alpha$ closer to 1, the steady-state MSD is lower while retaining comparable convergence rate. Therefore, $\alpha=1$ is preferred for the NR-NSAF algorithm.

Fig. 2 Mean behavior of the algorithm. (a) $\alpha = 0.5$, (b) $\alpha = 1$. [Gaussian input, SNR=10 dB, $\mu=0.5$, and $P=3$].

Fig. 4 depicts the MSD results of the NR-NSAF algorithm with $P=1$, 2, 3 and 4 values, where $P=1$ also denotes the NSAF algorithm. It is seen that theoretical calculation gives good fit with the simulation. Fig. 4(a) also shows that, in a low SNR scenario, the NR-NSAF algorithm has smaller steady-state MSD for larger $P$, without slowing convergence. In a high SNR case see Fig. 4(b), however, the convergence of the algorithm is slowed as $P$ increases. It follows that the NR-NSAF algorithm will work better than the NSAF algorithm when the environment is highly noisy.

Fig. 3 MSD curves of the algorithm versus $\alpha$ values. [Gaussian input, SNR = 10 dB, $\mu = 0.5$ and $P = 3$].

Fig. 4 MSD curves of the algorithm versus $P$ values. (a) SNR=10 dB, (b)
In Fig. 5, the MSD performance curves of the NR-NSAF algorithm with different step sizes ($\mu=0.5$, 0.1 and 0.4) are shown for both Gaussian and uniform inputs. Fig. 6 depicts similar results when the number of subbands is $N=4$. As one can see, the theoretical results fairly agree with the simulated results. It is worth noting that, as the member of constant step size based adaptive algorithms, users need to consider a compromise between fast convergence and low steady-state MSD when choosing the step size for the NR-NSAF algorithm.

4.2 Steady-state performance

Fig. 7 examines the effectiveness of (22) for predicting the steady-state MSD of the NR-NSAF algorithm as a function of the step size. The simulation values are the average over 200 MSDs at the steady-state stage. The step size $\mu$ is increased from 0.1 to 1. To fairly comparing with the theory presented in [24], we choose $\alpha=1$ and $\varepsilon=0$. As can be seen from Fig. 7, the proposed theoretical results have a good match with the simulated results for a smaller step size. However, a discrepancy of them can also be observed in Fig. 7 for larger step sizes. In comparison, the proposed (38) works better than the theory from [24] used for predicting the steady-state MSD of the NR-NSAF algorithm. This is because that the model in [24] requires large enough number of subbands and long adaptive filter.

5. Conclusion

We have analyzed in detail the performance of the NR-NSAF algorithm in terms of the transient and steady-state MSD. The proposed analysis is based on the vectorization and the Kronecker product of matrices, thereby it drops the specified distribution assumption of the input signal. In addition, the paraunitary assumption is unnecessary for the analysis filter banks in our analysis. For the special NSAF algorithm, the proposed steady-state expression outperforms the previous theory in [24] for a low-order adaptive filter scenario. Simulation results have shown good agreement with the theoretical results.

Appendix

In order to ensure the stability that the recursion (11) evolves with the iteration $k$, all the eigenvalues of the matrix $\Xi \triangleq \left(I_{MP} - \mu E \{A(k)\}\right) = \left[I_{M(P-1)} \ 0_{M(P-1)\times M}\right]$ must be inside the unit circle, i.e., $\rho(\Xi) < 1$. We rearrange $\Xi$ as

$$\Xi = \begin{bmatrix} \Xi_0 & \Xi_1 & \cdots & \Xi_{p-1} \\ \end{bmatrix},$$

where $\Xi_p = \beta_p \left(I_{M} - \mu E \{A(k)\}\right)$ for $p=0,1,...,P-1$. To proceed, we take advantage of the block-maximum-norm of the block matrix, with the notation $\|\cdot\|_{\infty}$ [29]:

$$\|\Psi\|_{\infty} \triangleq \max_{x \in \mathbb{R}^P} \|\Psi x\|_{\infty} / \|x\|_{\infty},$$

where $\Psi$ is an $MP \times MP$ matrix with block entries of size $M \times M$ each, and $x = [x_0, x_1, ..., x_{P-1}]^T$ an $MP \times 1$ vector with block entries of size $M \times 1$ each. Then, we obtain the following inequalities:
\[
\|\Xi\|_{\infty} = \max_{x \neq \mathbf{0}} \frac{\sum_{p=0}^{P-1} \|\Xi_p x_p\|_2}{\|x\|_\infty},
\]
\[
\leq \max_{x \neq \mathbf{0}} \frac{\sum_{p=0}^{P-1} \|\Xi_p x_p\|_2}{\|x\|_\infty},
\]
\[
\leq \max_{x \neq \mathbf{0}} \frac{\sum_{p=0}^{P-1} \|\Xi_p x_p\|_2}{\|x\|_\infty},
\]
\[
0 < \mu < \frac{2}{\lambda_{\text{max}}(E[A(k)])}.
\] (42)

6. References

[1] J. Benesty, and Y. Huang, Adaptive signal processing—applications to real-world problems, Berlin, Germany: Springer-Verlag, 2003.
[2] A. H. Sayed, Fundamentals of Adaptive Filtering. Hoboken, NJ, USA: Wiley, 2003.
[3] K. A. Lee, and W. S. Gan, “Improving convergence of the NLMS algorithm using constrained subband updates,” IEEE Signal Process. Lett., vol. 11, no. 9, pp. 736–739, 2004.
[4] J. Ni, and F. Li, “A variable step-size matrix normalized subband adaptive filter,” IEEE Trans. Audio Speech Lang. Process., vol. 18, no. 6, pp. 2648–2654, 2010.
[5] J. J. Jeong, K. Koo, G. T. Choi, and S. W. Kim, “A variable step size for normalized subband adaptive filters,” IEEE Signal Process. Lett. vol. 19, no. 12, pp. 906–909, 2012.
[6] H. J. Seo, and P. G. Park, “Variable individual step-size subband adaptive filtering algorithm,” Electron. Lett., vol. 50, no. 3, pp. 177–178, 2014.
[7] F. Yang, M. Wu, P. Ji, and J. Yang, “An improved multiband-structured subband adaptive filter algorithm,” IEEE Signal Process. Lett., vol. 19, no. 10, pp. 647–650, 2012.
[8] S. Y. Choi, S. E. Kim, and W. J. Song, “Noise-robust normalised subband adaptive filtering,” Electronics Letters, vol. 48, no. 8, 432–434, 2012.
[9] J. Ni, “Improved normalised subband adaptive filter,” Electron. Lett., vol. 48, no. 6, pp. 320–321, 2012.
[10] H. Zhao, Z. Zheng, Z. Wang, and B. Chen, “Improved affine projection subband adaptive filter for high background noise environments,” Signal Processing, 137: 356-362, 2017.
[11] Y. Yu, H. Zhao, R.C de Lamare, and L. Lu, “Sparsity-aware subband adaptive filters with adjustable penalties,” Digital Signal Processing, vol. 84, pp. 93-106, 2018.
[12] Y. Yu, H. Zhao, B. Chen, “Set-membership improved normalised subband adaptive filter algorithms for acoustic echo cancellation,” IET Signal Processing, vol. 12, no. 1, pp. 42-50, 2018.
[13] H. Cho, C. W. Lee, and S. W. Kim, “Derivation of a new normalized least mean squares algorithm with modified minimization criterion,” Signal Processing, vol. 89, no. 4, pp. 692–695, 2009.
[14] H. C. Shin, A. H. Sayed, “Mean-square performance of a family of affine projection algorithms,” IEEE Trans. Signal Process., vol. 52, no. 1, pp. 90–102, 2004.
[15] S. Zhang, J. Zhang, H. C. So, “Mean square deviation analysis of LMS and NLMS algorithms with white reference inputs,” Signal Processing, vol. 131, pp. 20-26, 2017.
[16] S. E. Kim, J. W. Lee, and W. J. Song, “A theory on the convergence behavior of the affine projection algorithm,” IEEE Trans. Signal Process., vol. 59, no. 12, pp. 6233–6239, 2011.
[17] S. E. Kim, J. W. Lee, and W. J. Song, “A noise-resilient affine projection algorithm and its convergence analysis,” Signal Process., vol. 121, pp. 1-4, 2016.
[18] M. S. Abadi, M. S. Shafiee, M. Zalaghi. “A low computational complexity normalized subband adaptive filter algorithm employing signed regressor of input signal,” EURASIP Journal on Advances in Signal Processing, vol. 2018, no. 1, pp. 1–23, 2018.
[19] K. A. Lee, and W. S. Gan, “Mean-square performance analysis of the normalized subband adaptive filter,” in Proc. 40th Asilomar Conf. Signals, Syst., Comput., 2006, pp. 248–252.
[20] J. Ni and X. Chen, “Steady-state mean-square error analysis of regularized normalized subband adaptive filters,” Signal Process., vol. 93, no. 9, pp. 2648–2652, 2013.
[21] W. Yin, and A. S. Mehr, “Stochastic analysis of the normalized subband adaptive filter algorithm,” IEEE Trans. Circuits Syst. I. Reg. Pap., vol. 58, no. 5, pp. 1020–1033, 2011.
[22] J. J. Jeong, S. H. Kim, G. Koo, and S. W. Kim, “Mean-square deviation analysis of multiband-structured subband adaptive filter algorithm,” IEEE Trans. Signal Process. vol. 64, no. 4, pp. 985–994, 2016.
[23] F. Yang, M. Wu, P. Ji, Z. Kuang, and J. Yang, “Transient and steady-state analyses of the improved multiband-structured subband adaptive filter algorithm,” IET Signal Process., vol. 9, no. 8, pp. 596–604, 2015.
[24] J. J. Jeong, K. Koo, G. Koo, and S. W. Kim, “Steady-state mean-square deviation analysis of improved normalized subband adaptive filter,” Signal Process., vol. 106, pp. 49–54, 2015.

[25] Y. Yu, H. Zhao, and L. Lu, “Steady-state behavior of the improved normalized subband adaptive filter algorithm and its improvement in under-modeling,” Signal, Image and Video Processing, vol. 12, no. 4, pp. 617–624, 2018.

[26] T. Y. Al-Naffouri and A. H. Sayed, “Transient analysis of adaptive filters with error nonlinearities,” IEEE Trans. Signal Process., vol. 51, no. 3, pp. 653-663, 2003.

[27] B. Chen, L. Xing, H. Zhao, N. Zheng, J.C. Principe, “Generalized correntropy for robust adaptive filtering,” IEEE Trans. Signal Process., vol. 64, no. 13, pp. 3376–3387, 2016.

[28] G. Alexander, Kronecker products and matrix calculus with applications. New York: Halsted, 1981.

[29] A. H. Sayed, “Diffusion adaptation over networks,” E-Reference Signal Processing, S. Theodoridis and R. Chellapa, Eds. Amsterdam, The Netherlands: Elsevier, 2013.

[30] R. A. Horn, C. R. Johnson, Matrix analysis, Cambridge university press, 2012.