IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 23, 2020, accepted October 23, 2020, date of publication October 29, 2020,
date of current version November 12, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3034694

Scheduled Step-Size Subband Adaptive Filter Algorithm With Implemental Consideration

TAESU PARK, (Student Member, IEEE), MINHO LEE, (Graduate Student Member, IEEE), AND POOGYEON PARK, (Senior Member, IEEE)

Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 790-784, South Korea

Corresponding author: Poogyeon Park (ppg@postech.ac.kr)

This work was supported in part by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea Government (MOTIE) (No. 20182020109350), and in part by the MSIT (Ministry of Science and ICT), Korea, under the ICT (IITP-2019-2011-1-00783).

ABSTRACT This article proposes a scheduled step-size normalized subband adaptive filter (NSAF) algorithm in stationary system environment. The mean-square deviation of the NSAF according to the step size is analyzed geometrically to construct a pre-designed trajectory. The mean-square deviation learning curve of the NSAF algorithm is forced to follow the pre-designed trajectory. This method removes the need for the NSAF algorithm to introduce tuning parameters and does not add any additional online computation. The table of the scheduled step sizes can be reconstructed online in proportion to not only the number of taps but also the number of subbands once they are scheduled offline. The novel memory-efficient scheduling scheme minimizes the memory space required and simplifies operation without performance degradation. Because of these features, the proposed algorithm performs as well as the variable-step-size NSAFs studied previously, and is very suitable for chip level implementation in terms of computational complexity and memory space. Simulation results show that the proposed algorithm is robust against external environment change and has good performance compared to the existing variable step-size algorithms without any additional online computation and tuning parameter.

INDEX TERMS Adaptive filters, normalized subband adaptive filter algorithm, scheduled step size.

I. INTRODUCTION

Adaptive filters are widely used in signal processing fields such as system identification, echo cancellation, and active noise control [1]–[5]. The primary performance indicator of a good adaptive filter is that the algorithm should have fast convergence speed, low steady-state error, and low computational complexity. Among the adaptive filter algorithms, the normalized-least-mean-square (NLMS) algorithm is most commonly used because of its simplicity of implementation and low computational complexity [6]–[8]. However, it suffers a large performance degradation in terms of convergence speed for correlated inputs. To overcome this drawback, several adaptive filter algorithms have been proposed. The affine-projection (AP) algorithm is one of an important family of adaptive filter algorithms [9]–[12]. It can compensate for the disadvantage of the NLMS algorithm by accumulating and using multiple input vectors instead of one input vector, but the computational complexity increases because of the matrix inverse operation. To solve this computational complexity problem, a method of updating the auxiliary filter coefficient instead of updating the actual adaptive filter coefficient has been proposed [13]–[17]. This method uses the time-shift structure of the input signal to represent the actual adaptive filter coefficient as the sum of the auxiliary coefficients. The advantage is that when only the error signal is needed, updating this auxiliary coefficient is sufficient. An approximate filtering method using only the first element of the error vector and approximating values using the first element for the remaining elements was also studied [13], [14]. This method reduced the computational complexity from MK to M, where K denotes for the projection order. Because the approximate filtering method includes assumptions, a fast AP algorithm without assumptions has also been proposed [18], [19]. Many other studies on the AP algorithm are summarized in [20].

The normalized subband adaptive filter (NSAF) algorithm is another important family of algorithms that can improve the convergence performance of adaptive filters [3], [21]–[23]. Lee and Gan proposed the NSAF, which is based on the minimum disturbance principle [23]. The NSAF algorithm has the effect of whitening the correlated input

The associate editor coordinating the review of this manuscript and approving it for publication was Halil Ersin Soken.

because it divides the input signal and the desired signal by several frequency bands, downsamples it, and reconstructs it. However, the performance of NSAF is severely degraded when impulsive noise is introduced. Impulsive noise refers to noise having a very large amplitude in a short time with a low probability. This impulsive noise environment is found in various fields such as acoustic signal processing, active noise control, and communication. To secure robustness against such impulsive noise, the sign NSAF algorithm and step-size scaler concept were proposed [24], [25]. The sign NSAF algorithm mitigates the performance degradation caused by impulsive noise by using only the sign of the error signal, but has a disadvantage in that the convergence speed is slow. The step-size scaler does not update the filter coefficients by making the step size very small when impulsive noise occurs.

In addition to the research to mitigate the performance degradation of the NSAF algorithm caused by the external environment, research to improve its inherent performance was also conducted. The step size of the NSAF algorithm is a trade-off between the convergence rate and the steady-state error. To address this problem, several variable-step-size NSAFs were introduced [26]–[31]. A variable-step-size matrix NSAF (VSSM-NSAF) was proposed by making the assumption that the noise variance is the same as the L_2-norm of the *a posteriori* error [26]. The VSSM-NSAF algorithm exhibits good tracking performance in a non-stationary environment by estimating the power of system noise, but still suffers from a large steady-state error. The NSAF with variable step size (NSAF-VSS) calculates the step size at every iteration [27]. Because it calculates the optimal step size for each iteration, it exhibits a faster convergence rate and lower steady-state error compared to a fixed-step-size NSAF. However, because NSAF-VSS does not estimate the variance of the error signal, it does not respond immediately to sudden system changes. To solve this problem, the variable step-size NSAF (VSS-NSAF) was proposed [28]. Because the VSS-NSAF estimates the variance of the error signal at every iteration, the algorithm converges well even if a sudden system change occurs. However, VSS-NSAF requires an additional online calculation to estimate the error signal. The variable individual step-size NSAF (VISS-NSAF) further improves the performance of the adaptive filter by calculating the optimal step size for each frequency band for each iteration [29], [30]. However, it also requires an online calculation to compute the optimal step size for each subband. Recently, the novel variable-step-size NSAF (NVSS-NSAF) was proposed by using a combined error cost function, but it did not exhibit better performance than the previously studied variable-step-size NSAFs [31].

These variable-step-size NSAFs have much better convergence performance than fixed-step-size NSAFs, but require several tuning parameters and additional online calculations to compute the optimal step size in each iteration. These tuning parameters are difficult to set and should be reset if the system environment changes. Further, the speed of the algorithm decreases due to the increase in the computational complexity. To solve this problem, many studies have been conducted to reduce the computational amount of NSAF [32]–[34]. Although algorithms to reduce the computational amount have been introduced, algorithms that include the variable-step-size concept still cannot avoid additional online computation. To completely solve this problem, the step-size scheduling scheme was studied for NLMS and AP [35], [36]. By geometrically analyzing the mean-square-deviation (MSD) learning curve, a step-size table associated with a pre-designed MSD curve is created and assigned every moment. This approach exhibits very good convergence performance without involving any tuning parameters or additional online calculations. However, this approach has not been studied for NSAF, one of the important families of adaptive filters. The NLMS and AP algorithms are not suitable for actual chip-level implementation because the performance of the NLMS algorithm is highly degraded for correlated input, and the AP algorithm involves the matrix inversion operation.

Therefore, this article proposes a scheduled step-size NSAF (SS-NSAF) algorithm considering the actual chip-level implementation of the algorithm. In this study, the step-size scheduling scheme is extended to the NSAF in a stationary environment so as to obtain excellent performance without additional tuning parameters or online calculations. A new MSD analysis is conducted for a fixed step size, and based on this, the optimal step size in each iteration is found and stored in a table. In addition, it was proved that the proposed algorithm can be automatically adapted to the number of taps and the number of subbands. The scheduled step-size table forces the MSD curve to follow the pre-designed trajectory consisting of an iteration and step size pair, without any additional online calculation, thereby providing excellent convergence performance. In addition, we propose a novel memory-efficient step-size scheduling method to minimize the memory space required by the algorithm and maximize the operational efficiency, thus suggesting an optimal algorithm for real chip-level implementation.

This article is organized as follows. Section 2 analyzes the MSD to find the two asymptotes of the NSAF algorithm, and Section 3 schedules the step size based on this. Section 4 proposes a modified scheduling method for memory efficiency and chip-level implementation efficiency. Finally, Section 5 evaluates the performance of the proposed algorithm by comparing it to that of other algorithms.

II. MSD ANALYSIS OF THE NSAF ALGORITHM

The desired signal $d(n)$ is defined as

$$d(n) = \mathbf{u}^T(n)\,\mathbf{w} + v(n), \tag{1}$$

where $\mathbf{u}(n)$ is M-dimensional input vector with variance σ_u^2, and $\mathbf{w}$ is an M-dimensional optimal weight vector that we have to estimate. The variable $v(n)$ is white Gaussian measurement noise with variance σ_v^2, and it is independent of the input signal.

Fig. 1 shows a block diagram of the NSAF algorithm, where N is the number of subbands, $\mathbf{A}_j(z), j = 0, \ldots, N-1$ are analysis filters, and $\mathbf{S}_j(z), j = 0, \ldots, N-1$ are

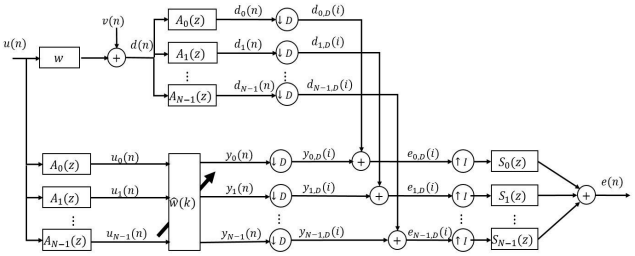

FIGURE 1. Block diagram of the NSAF algorithm.

synthesis filters. The desired signal $d(n)$ and the input signal $\mathbf{u}(n)$ are divided into frequency bands by the analysis filters. Each divided signal is defined as $d_j(n)$ and $\mathbf{u}_j(n)$, where $j = 0, 1, \ldots, N-1$. The adaptive filter output is $y_j(n)$. The partitioned signals $d_j(n)$ and $y_j(n)$ are then decimated by a factor D. These two decimated signals, $d_j(n)$ and $y_j(n)$, are added together and then interpolated with the interpolation factor I. The synthesis filter bank $\mathbf{S}(z)$ then merges these interpolated signals. The decimated output signal is defined as $y_{j,D}(i) = \mathbf{u}_j^T(i)\hat{\mathbf{w}}$, where $\mathbf{u}_j(i) = [u_j(iN), u_j(iN-1), \ldots, u_j(iN-M+1)]^T$, and $\hat{\mathbf{w}}(i)$ is an estimate of $\mathbf{w}$ at the ith iteration. The index n denotes the time index of the original signal, and the index i denotes the time index of the decimated signal. In this study, D and I are set equal to N for critical decimation.

The basic adaptive filter coefficient update equation of the NSAF algorithm can be written as

$$\hat{\mathbf{w}}(i+1) = \hat{\mathbf{w}}(i) + \mu \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)} e_{j,D}(i). \tag{2}$$

The weight-error vector of the system is defined as $\tilde{\mathbf{w}}(i) \triangleq \mathbf{w} - \hat{\mathbf{w}}(i)$, then the basic adaptive filter coefficient update equation, (2), is expressed as

$$\begin{aligned}\tilde{\mathbf{w}}(i+1) &= \tilde{\mathbf{w}}(i) - \mu \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)} e_{j,D}(i) \\ &= \mathbf{F}(i)\,\tilde{\mathbf{w}}(i) - \mu \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)} v_{j,D}(i), \end{aligned} \tag{3}$$

where $\mathbf{F}(i) = \mathbf{I}_M - \mu \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)\mathbf{u}_j^T}{\mathbf{u}_j^T(i)\mathbf{u}_j(i)}$, and $\mathbf{I}_M$ is an M-by-M identity matrix. The measurement noise signal $v_{j,D}(i)$ is a white Gaussian signal with average 0 and variance $\sigma_{v_j,D}$. To perform MSD analysis, we define the MSD as $MSD(i) \triangleq E\left(\tilde{\mathbf{w}}^T(i)\,\tilde{\mathbf{w}}(i) \mid \mathcal{U}_i\right) = Tr(\mathbf{P}(i))$, where $\mathbf{P}(i) \triangleq E\left(\tilde{\mathbf{w}}(i)\,\tilde{\mathbf{w}}^T(i) \mid \mathcal{U}_i\right)$ and $\mathcal{U}_i \triangleq \{\mathbf{u}_k \mid 0 \le k < i\}$. We consider $\mathbf{u}(i)$ as a deterministic quantity for $\mathbf{P}(i)$. The weight-error vector $\tilde{\mathbf{w}}(i)$ and the measurement noise signal $v_{j,D}(i)$ are assumed to be independent of each other. The relation between $\mathbf{P}(i)$ and $\mathbf{P}(i+1)$ can then be obtained as follows:

$$\begin{aligned}\mathbf{P}(i+1) &= \mathbf{F}(i)\,\mathbf{P}(i)\,\mathbf{F}^T(i) \\ &\quad + \mu^2 \sum_{j=0}^{N-1} \sigma_{v_j,D}^2 \frac{\mathbf{u}_j(i)\,\mathbf{u}_j^T(i)}{\left(\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)\right)^2}. \end{aligned} \tag{4}$$

By taking the trace operation on both sides of (4), the following equation can be obtained:

$$\begin{aligned}Tr(\mathbf{P}(i+1)) &= Tr(\mathbf{F}^T(i)\,\mathbf{F}(i)\,\mathbf{P}(i)) \\ &\quad + \mu^2 \sum_{j=0}^{N-1} \sigma_{v_j,D}^2 \frac{\mathbf{u}_j(i)\,\mathbf{u}_j^T(i)}{\left(\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)\right)^2}. \end{aligned} \tag{5}$$

Now, $Tr(\mathbf{F}^T(i)\,\mathbf{F}(i)\,\mathbf{P}(i))$ can be arranged and bounded as follows.

$$\begin{aligned}&Tr(\mathbf{F}^T(i)\,\mathbf{F}(i)\,\mathbf{P}(i)) \\ &= Tr(\mathbf{P}(i) - 2\mu \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)\,\mathbf{u}_j^T(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)}\mathbf{P}(i) \\ &\quad + \mu^2 \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)\,\mathbf{u}_j^T(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)}\mathbf{P}(i)) \\ &= Tr(\mathbf{P}(i)) + \left(-2\mu + \mu^2\right) \sum_{j=0}^{N-1} \frac{\mathbf{u}_j^T(i)\,\mathbf{P}(i)\,\mathbf{u}_j(i)}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)} \\ &\le Tr(\mathbf{P}(i)) + \left(-2\mu + \mu^2\right) N\lambda_{min}(\mathbf{P}(i)), \end{aligned} \tag{6}$$

where $\lambda_{min}(\mathbf{P}(i))$ is the minimum eigenvalue of the MSD matrix $\mathbf{P}(i)$. Because the inequality $M\lambda_{min}(\mathbf{P}(i)) \le Tr(\mathbf{P}(i))$ is always satisfied, there exists some $\beta \ge 1$, which gives the approximation $\lambda_{min}(\mathbf{P}(i)) \approx \frac{Tr(\mathbf{P}(i))}{\beta M}$. By combining this approximation with (6), (5) can be written as

$$\begin{aligned}Tr(\mathbf{P}(i+1)) &\approx \left(-2\mu + \mu^2\right) N\lambda_{min}(\mathbf{P}(i)) \\ &\quad + Tr(\mathbf{P}(i)) + \mu^2 \sum_{j=0}^{N-1} \frac{\sigma_{v_j,D}^2}{\mathbf{u}_j^T(i)\,\mathbf{u}_j(i)} \\ &= \gamma Tr(\mathbf{P}(i)) + \delta. \end{aligned} \tag{7}$$

For notational convenience, two variables γ and δ are defined as $\gamma \overset{\text{def}}{=} \left(1 - \left(2N\mu - N\mu^2\right)/(\beta M)\right)$, $\delta \overset{\text{def}}{=} \mu^2 \sum_{j=0}^{N-1} \frac{\sigma_{v_j,D}^2}{\mathbf{u}_j^T(i)\mathbf{u}_j(i)}$. The general term of the recursion formula (7) can then be obtained as follows:

$$Tr(\mathbf{P}(i)) \approx \gamma^i Tr(\mathbf{P}(0)) + \sum_{k=0}^{i} \gamma^k \delta. \tag{8}$$

$Tr(\mathbf{P}(i))$ in (8) consist of two parts, the transient part $h_1(i,\mu) \triangleq \gamma^i Tr(\mathbf{P}(0))$ and the steady-state part $h_2(i,\mu) \triangleq \sum_{k=0}^{i} \gamma^k \delta$:

$$h_1(i,\mu) \overset{\text{def}}{=} \gamma^i Tr(\mathbf{P}(0)), \quad h_2(i,\mu) \overset{\text{def}}{=} \sum_{k=0}^{i} \gamma^k \delta. \tag{9}$$

To select the optimum β, we recommend comparing the MSD learning curve to a wide range of $\beta \ge 1$ for a fixed step size. The most important element in the system is the nature of the input. For example, if the input signal is white Gaussian, the beta must be 1.

III. SCHEDULED STEP-SIZE NSAF ALGORITHM

The proposed algorithm is obtained by geometrically analyzing the MSD curve of the NSAF algorithm. For a fixed step size between 0 and 1, $h_1(i,\mu)$ and $h_2(\infty,\mu)$ defined in (9)

produce straight lines, respectively. The MSD learning curve switched from the transient part to the steady-state part near the intersection of the two straight lines. As in [35], these intersections can uniquely defined for any fixed step size, which is proved in the following theorem.

Theorem 1: For any $i_R > i_0$, where i_0 is included in the set of all positive and zero integers such that

$$i_0 \triangleq \arg\min_i \left[\left(i - \frac{N}{\beta M} \right)^i Tr\left(\mathbf{P}\left(0\right)\right) \leq \frac{\beta\mu}{2-\mu} \frac{\sigma_v^2}{\sigma_x^2} \right] \tag{10}$$

there exists a unique $\mu \in (0, 1]$, say $f_{N,M}(i_R)$, that satisfies $h_1(i_R, \mu) = h_2(\infty, \mu)$.

Proof 1: As the step size μ increases from 0 to 1, $h_1(i_R, \mu)$ strictly decreases, but $h_2(\infty, \mu)$ strictly increases. For any $i_R > i_0$, $h_1(i_R, 0) > h_2(\infty, 0)$ and $h_1(i_R, 1) < h_2(\infty, 1)$, which proves the theorem.

For a fixed step size μ, the point where $h_1(i_R, \mu)$ and $h_2(\infty, \mu)$ intersect, i.e., the unique time for the MSD curve to reach steady-state part, can be obtained. After that cross point, it is difficult to expect that the MSD will decrease any further, and therefore the step size must be changed if the MSD curve reaches this cross point. With this scheme, the step size μ can be scheduled as follows:

$$\mu_i \triangleq \begin{cases} 1, & \text{for } i \leq i_0 \\ f_{N,M}(i), & \text{for } i > i_0, \end{cases} \tag{11}$$

where $f_{N,M}(\cdot)$ is defined in Theorem 1. Whereas the real numbers i_R, the intersection of $h_1(i_R, \mu)$ and $h_2(\infty, \mu)$, can be calculated for any real number $\mu \in (0, 1]$, the proposed algorithm has to generate μ for all integers $i > i_0$, which is analytically impossible. However, Theorem 1 guarantees the one-to-one correspondence of $f_{N,M} : \{i | i \in [i_0, \infty)\} \rightarrow \{\mu | \mu \in (0, 1]\}$, which makes it possible to construct the step-size table by using the following inverse function:

$$f_{N,M}^{-1}(\mu) = \frac{\log\left(\frac{h_2(\infty,\mu)}{Tr[\mathbf{P}(0)]}\right)}{\log\left(1 - \frac{2N\mu - N\mu^2}{\beta M}\right)}. \tag{12}$$

Fig. 2 helps intuitively understand how to change the step size when moving from the current iteration to the next iteration. In addition, Table 1 summarizes the scheduled step-size NSAF algorithm described above.

The following lemma shows the scalability of the constructed step-size table with respect to the number of taps M and the number of subbands N, which means that it is easy to adapt the constructed lookup table to the number of taps and the number of subbands.

Lemma 1: For arbitrary tap lengths M_1 and M_2, and the number of subbands N_1 and N_2, the function $f_{N,M}^{-1}(\cdot)$ satisfies

$$\frac{f_{N_1,M_1}^{-1}(\mu)N_1}{M_1} \approx \frac{f_{N_2,M_2}^{-1}(\mu)N_2}{M_2}. \tag{13}$$

Proof 2: $h_1(i, \mu)$ in (9) can be approximated as

$$h_1(i, \mu) \simeq \left(1 - \frac{2\mu - \mu^2}{\alpha}\right)^{\frac{\alpha N}{\beta M} i} Tr\left(\mathbf{P}(0)\right), \tag{14}$$

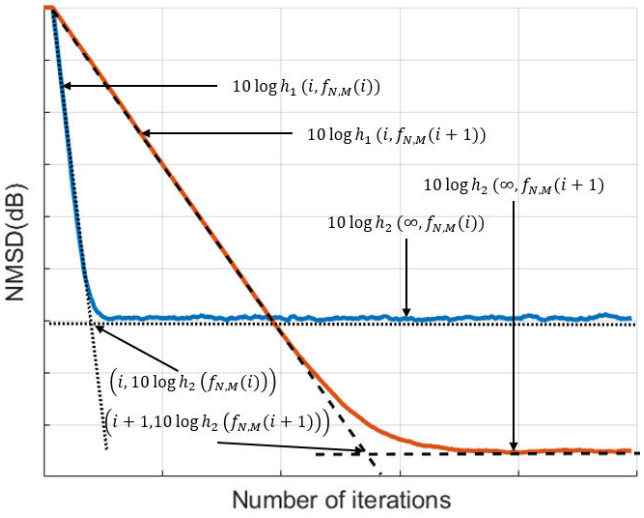

FIGURE 2. Determine step size at the *i*th iteration.

TABLE 1. Scheduled step-size NSAF algorithm summary.

1. Initialization.

$$Tr\left(\mathbf{P}(0)\right) = 0, \beta = 1, \mu_0 = 1$$

2. Step-size table construction.

2-1. Construct pairs of (i_R, μ),

$$\mathcal{S}_n = \left\{ \left[f_{N,M}^{-1}(\mu), \mu \right] \mid \mu = \frac{q}{r}, q = 1, ..., r \right\},$$

where r is a positive integer.

2-2. Interpolate the pairs of $\mathcal{S}_n$.

Find the maximum value $i_{R,M}$ and the minimum value $i_{R,m}$ that satisfy $i_{R,m} \leq i < i_{R,M}$ such that $(i_{R,m}, \mu_m) \in \mathcal{S}_n$ and $(i_{R,M}, \mu_M) \in \mathcal{S}_n$ and take

$$f_{N,M}(i) = \mu_m + \frac{(i - i_{R,M})}{i_{R,m} - i_{R,M}}(\mu_m - \mu_M).$$

2-3. Using the inverse interpolation method of 2-2, $\left(i, f_{N,M}(i)\right)$ pairs are obtained for all integers i.

3. Change the step size according to the obtained (i, μ_i) pairs.

```
for each i do
    if i ≤ i_o
        μ_i = 1
    else
        μ_i = f_{N,M}(i)
    endif
endfor
```

where α is any constant satisfying the condition $(2\mu - \mu^2)/(\alpha) \ll 1$ and $(\alpha N)/(\beta M) \gg 1$. Then (12) becomes

$$f_{N,M}^{-1}(\mu) \approx \frac{\beta M}{\alpha N} \frac{\log\left(\frac{h_2(\infty,\mu)}{Tr(\mathbf{P}(0))}\right)}{\log\left(1 - \frac{2\mu - \mu^2}{\alpha}\right)}. \tag{15}$$

Therefore, once we construct the step-size lookup table for the given tap length M_1 and the number of the subbands N_1,

TABLE 2. Reset algorithm for sudden system change.

$i_{base} = 0, \bar{\sigma}^2_{e,0} = \sigma^2_d$

$e^2_{th} = \frac{2+(\beta-\mu)}{2-\mu}\sigma^2_v$
for each i **do**
  $\bar{\sigma}^2_{e,i+1} = \bar{\sigma}^2_{e,i+1} = \alpha\bar{\sigma}^2_{e,i} + (1-\alpha)e^2_i$
  if $\bar{\sigma}^2_{e,i} > \gamma e^2_{th}$ **then**
    $i_{base} = i, \bar{\sigma}^2_{e,0} = \sigma^2_d$
  endif
  $\hat{\mathbf{w}}(i+1) = \hat{\mathbf{w}}(i) + \mu_{i-i_{base}} \sum_{j=0}^{N-1} \frac{\mathbf{u}_j(i)}{\mathbf{u}^T_j(i)\mathbf{u}_j(i)} e_{j,D}(i)$
endfor

the table can be easily modified for any M_2 and N_2 by using this scalability property.

IV. PRACTICAL CONSIDERATIONS

A. STEP-SIZE SCHEDULING SCHEME

The step-size lookup table for every iteration occupies a large memory space. In addition, when an algorithm is actually implemented at the chip level, the division operation performs an inaccurate approximation operation. To solve these practical problems, the following downsampled scheduling method is proposed. The proposed method constructs the lookup table with only m pairs of $(i^*(k), \mu^*(k))$ for $k = 1, \ldots, m$:

$$0.5\mu^*(k-1) = \mu^*(k), \quad i^*(k) \triangleq \lfloor f^{-1}_{N,M}(\mu^*(k)) \rfloor - 1 \tag{16}$$

where $i^*(0) \triangleq i_0$ and $\mu^*(0) \triangleq 1$. By changing the step size to half of the previous value at $i^*(k)$, if only one initial step size is stored in the memory space in a real chip implementation situation, the step size can be changed at the scheduled time through only floating-point arithmetic.

B. SUDDEN SYSTEM CHANGES: RESET ALGORITHM OF THE STEP-SIZE TABLE

Because the proposed algorithm is designed for a stationary environment, a reset algorithm is needed to cope with sudden system changes. The reset algorithm used in [36] is applied to the proposed algorithm. The reset algorithm is based on the error signal, where $\bar{\sigma}^2_{e,i}$ is the time-averaged value of e^2_i. With $\mathbf{w} = 0$, $\bar{\sigma}^2_{e,i}$ is calculated as

$$\bar{\sigma}^2_{e,0} = E(e^2_0) = \sigma^2_d, \tag{17}$$

$$\bar{\sigma}^2_{e,i+1} = \alpha\bar{\sigma}^2_{e,i} + (1-\alpha)e^2_i. \tag{18}$$

The covariance of the *a priori* measurement error e_i converges as follows:

$$\begin{aligned}\lim_{i\to\infty} E(e^2_i) &\approx \sigma^2_u Tr(\mathbf{P}(\infty)) + \sigma^2_v \\ &\approx \frac{2+(\beta-\mu)}{2-\mu}\sigma^2_v \triangleq e^2_{th}.\end{aligned} \tag{19}$$

If $\bar{\sigma}^2_{e,i} > \gamma e^2_{th}$, it is determined that the system has changed, and the scheduling sequence is initialized. The reset algorithm is summarized in Table 2.

C. COMPUTATIONAL COMPLEXITY

The number of computations of the proposed algorithm is compared with other variable step-size algorithms. The length of $\mathbf{w}$ is M, the lengths of the analysis filter bank and the synthesis filter bank are both L, and the number of subbands is N. The computational complexities of the SS-NSAF and other existing VSS-NSAF algorithms are summarized in Table 3. A typical NSAF algorithm requires $3M + 3NL + 1$ online computation for every iteration. NSAF-VSS algorithm requires more computation to calculate the optimal step size for each iteration, which requires on-line computation as much as $2M + 14$. So the total computational complexity of the NSAF-VSS algorithm is $5M + 3NL + 15$. Also, the VSS-NSAF algorithm requires a total amount of $5M + 3NL + 4N + 14$ to estimate the error signal at every iteration and to calculate the optimal step size based on this error signal. NVSS-NSAF algorithm also requires the calculation of $3M + 3NL + N(M + 2N + 7)3M + 3NL + 7$ because of the additional estimation process and the calculation of optimal individual step size for each iteration. On the other hand, the proposed algorithm does not require any additional multiplication operation because the step-size table is calculated off-line and the pre-stored step sizes are applied in every iteration in the on-line process.

TABLE 3. Summary of the computational complexity.

Algorithm	Computational Complexity
NSAF	$3M + 3NL + 1$
NSAF-VSS	$5M + 3NL + 15$
VSS-NSAF	$5M + 3NL + 4N + 14$
NVSS-NSAF	$3M + 3NL + N(M + 2N + 7)3M + 3NL + 7$
Proposed algorithm	$3M + 3NL + 1$

V. SIMULATION RESULTS

In this section, various simulations are presented to verify the performance of the proposed algorithm in the channel estimation scenario. The unknown filter coefficient $\mathbf{w}$ was generated randomly as a unit vector, and the length of the adaptive filter and the length of the unknown filter were set to be the same. Also $Tr(\mathbf{P}(0))$ is assumed to be zero. The signal-to-noise ratio (SNR) is defined as

$$SNR \triangleq 10log_{10}\left(\sigma^2_y/\sigma^2_v\right). \tag{20}$$

In order to verify the performance of the proposed algorithm in a correlated input situation, the autoregressive (AR) models are set as follows in equations (21), (22) and the white Gaussian input is filtered and applied to the simulation environment.

$$R_1(z) = \frac{1}{1-0.95z^{-1}}. \tag{21}$$

$$R_2(z) = \frac{1}{1-1.6z^{-1}+0.81z^{-2}}. \tag{22}$$

In order to confirm how well the proposed algorithm tracks the changed system when there is a sudden change in the system environment at a certain point in time, we have newly generated $\mathbf{w}$ at half of the total iterations. In addition, a simulation was conducted to show that the proposed algorithm is robust in a system environment mismatch situation. Each simulation result was obtained through 10 independent trials.

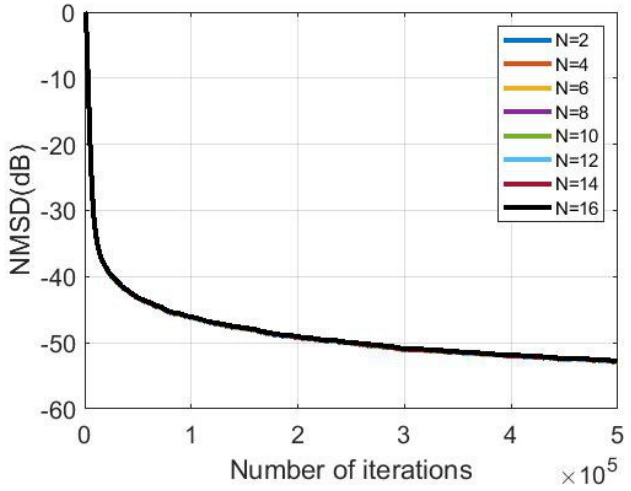

FIGURE 3. NMSD learning curves of the proposed SS-NSAF algorithm according to the number of subbands N (M=1024, SNR=30dB, White Gaussian input).

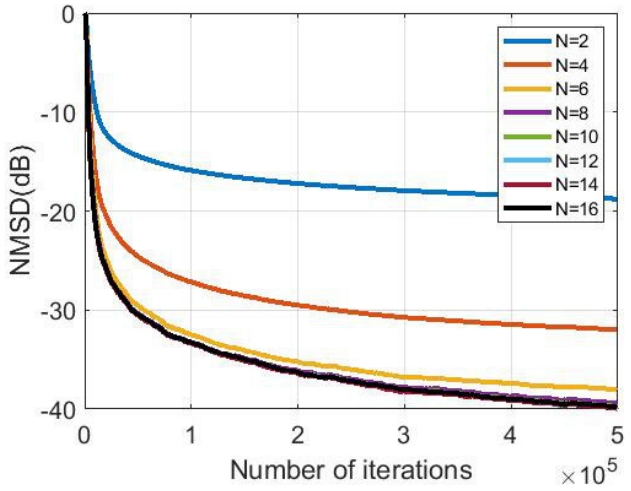

FIGURE 4. NMSD learning curves of the proposed SS-NSAF algorithm according to the number of subbands N (M=1024, SNR=30dB, AR_1 input).

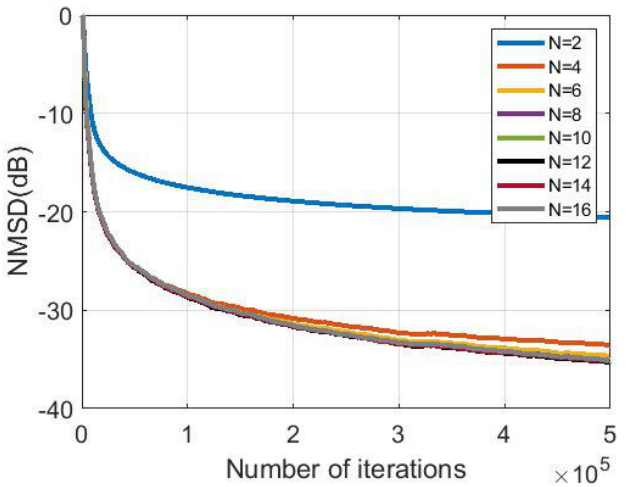

FIGURE 5. NMSD learning curves of the proposed SS-NSAF algorithm according to the number of subbands N (M=1024, SNR=30dB, AR_2 input).

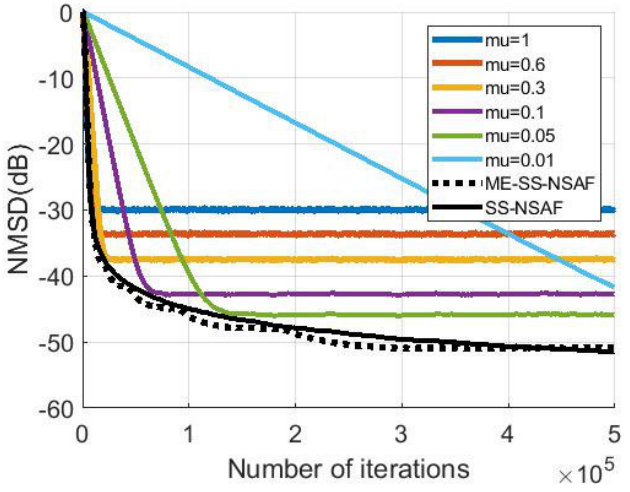

FIGURE 6. NMSD learning curves of the conventional-NSAF, ME-SS-NSAF and SS-NSAF algorithm (M=1024, N=8, SNR=30dB, White Gaussian input).

A. INPUT SELF-WHITENING EFFECT THROUGH THE NUMBER OF SUBBANDS

For correlated input signals, a proper choice of the number of subbands N should be made. In Fig. 3, 4, 5, M and SNR are set to 1024 and 30dB, respectively. The length of the prototype filters are set to 8 times N.

The MSD curves according to the number of subbands in a white Gaussian input situation are shown in Fig. 3. Since the input signal is not correlated, the NSAF's self-whitening effect does not work in this case. Therefore, there is almost no difference in convergence performance of the MSD curves according to the number of subbands. The MSD learning curves according to the number of subbands in AR_1 input environment is shown in Fig. 4. In this case, the best performance is achieved when $N = 14$. This means that the input signal is sufficiently whitened at $N = 14$. Fig. 5 shows the MSD learning curves according to the number of subbands for the AR_2 input. In this case, the misalignment of the proposed algorithm decreases as the number of subbands increases. The proposed algorithm shows the best performance when about $N = 14$. This indicates that the input signal is sufficiently whitened when $N = 14$.

B. PERFORMANCE COMPARISON BETWEEN SCHEDULED STEP-SIZE NSAF AND FIXED STEP-SIZE NSAF

The following simulation in Fig. 6 was performed to check how well the MSD curves of the SS-NSAF algorithm follows the pre-designed trajectory generated from the respective fixed step-size NSAF algorithms. The simulation was performed in the environment of $M = 1024$, $N = 8, SNR = 30dB$. The simulation result shows that SS-NSAF follow the pre-designed trajectory well. Therefore, it was confirmed that the proposed algorithm closely follows the pre-designed trajectory guaranteed by each step size for all step sizes between 0 and 1 without any additional online calculation.

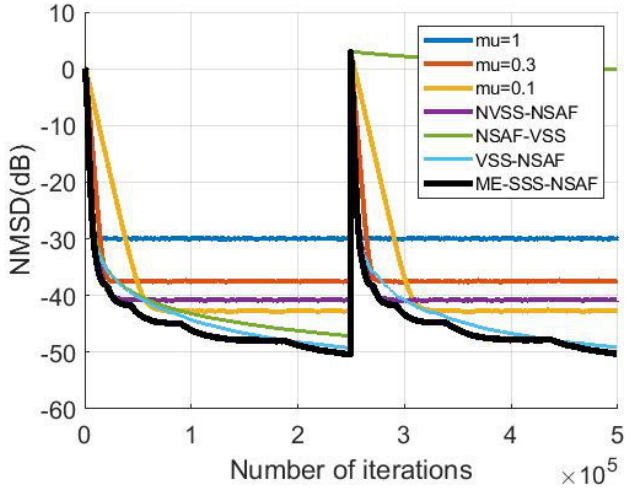

FIGURE 7. NMSD learning curves of the conventional-NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and proposed algorithm with sudden system change environment. (M=1024, N=8, SNR=30dB, White Gaussian input).

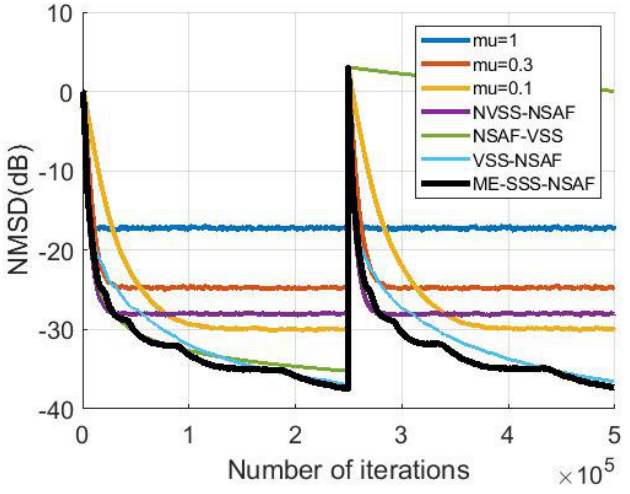

FIGURE 8. NMSD learning curves of the conventional-NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and proposed algorithm with sudden system change environment. (M=1024, N=8, SNR=30dB, AR_1 input).

C. NOVEL MEMORY EFFICIENT SCHEDULING

The black dotted line in Fig. 6 represents the MSD learning curve of the novel memory efficient scheduled step-size NSAF (ME-SS-NSAF). As can be seen in the Fig. 6, ME-SS-NSAF shows almost the same performance as the full-table version. The newly proposed ME-SS-NSAF algorithm maximizes the efficiency of memory space by storing only the initial step-size value. In addition, by reducing the step size by half at the next step-size change point, it guarantees almost the same performance as the full table version of the SS-NSAF with a simple decimal point shift operation in chip-level implementation.

D. PERFORMANCE COMPARISON

In this simulation, the performance of the proposed algorithm is compared with that of the existing algorithms. The novel variable-step-size NSAF (NVSS-NSAF) designed by Wen and Zhang [31], the NSAF with variable-step-size (NSAF-VSS) introduced by Shin *et al.* [27] and the variable-step-size NSAF (VSS-NSAF) proposed by Jeong *et al.* [28] are included. The simulations were performed to verify the performance of the proposed algorithm for various conditions. The number of subbands N is set to be two cases, 8 and 16, and the length of the proto type filter is set to be N^2. $Q = 1, \delta = 0.0001$ and $\lambda = 0.9945$ for the NVSS-NSAF, Tr [$\mathbf{P}(0)$] is set to be 100 for the NSAF-VSS, $J(0)$ is set to be 1 for VSS-NSAF, Tr [$\mathbf{P}(0)$] $= 0$ and $\beta = 1$ for the proposed algorithm.

Fig. 7, 8, 9 show the MSD learning curves of the conventional NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and the proposed algorithm for white Gaussian input, $AR(1)$ input and $AR(2)$ input. As can be seen in Fig. 7, 8, 9, the proposed algorithm shows very similar performance to other existing algorithms. The variable-step-size NSAF papers compared with the proposed algorithm have very good performance in terms of convergence speed and steady-state error. NVSS-NSAF is a recently proposed algorithm, but its performance was

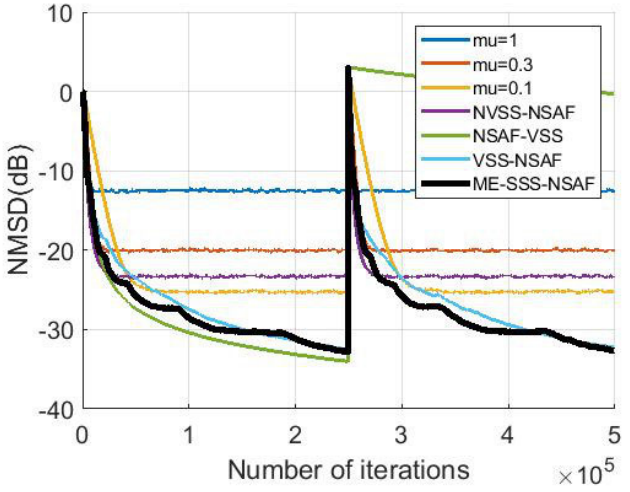

FIGURE 9. MSD convergence performance comparison between conventional-NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and proposed algorithm with sudden system change environment. (M=1024, N=8, SNR=30dB, AR_2 input).

not very good. NSAF-VSS calculates the optimal step size for every iteration through MSD analysis, but when the system changes significantly, NSAF-VSS did not converge. VSS-NSAF also calculates the optimal step size for every iteration. VSS-NSAF also estimated the variance of the error signal as well, showing the ability to converge well even with sudden system changes. However, since all the compared papers calculate optimal step size in every iteration, the additional online calculation increases greatly. On the other hand, it can be confirmed that the proposed algorithm shows similar performance to variable-step-size NSAF algorithms that perform very well without any additional online calculations, and also shows good performance in sudden system changes due to the application of the reset algorithm. In addition, as shown in Fig. 10 and Fig. 11, the proposed algorithm can reconstruct the step-size table by scalability property even if the tap length or the number of subbands is different.

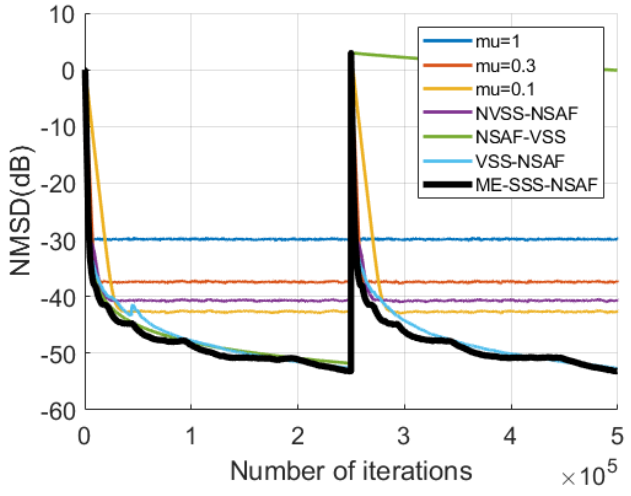

FIGURE 10. MSD convergence performance comparison between conventional-NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and proposed algorithm with sudden system change environment. (M=512, N=8, SNR=30dB, White Gaussian input).

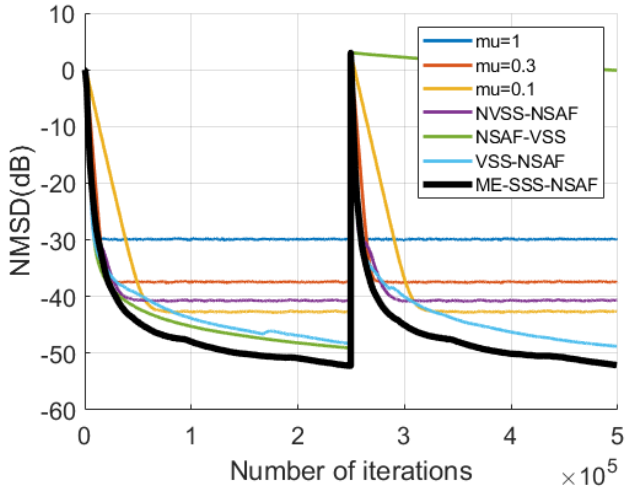

FIGURE 11. MSD convergence performance comparison between conventional-NSAF, NVSS-NSAF, NSAF-VSS, VSS-NSAF and proposed algorithm with sudden system change environment. (M=1024, N=16, SNR=30dB, White Gaussian input).

Even in these various environments, it was confirmed that the proposed algorithm still shows good performance.

E. INCORRECTLY ESTIMATED ENVIRONMENT

Fig. 12 shows the proposed algorithm's robustness against the incorrectly estimated SNR. This simulation shows the MSD learning curves of the proposed algorithm and the existing algorithms when the actual SNR= $50dB$ is incorrectly estimated by SNR= $20dB$. Since the NSAF-VSS algorithm does not estimate the power of the measurement noise σ_v^2, the performance is significantly reduced if the SNR is incorrectly estimated. On the other hand, the VSS-NSAF algorithm shows robust performance in the incorrectly estimated SNR environment. However, the VSS-NSAF algorithm estimates the power of the measurement noise in every iteration, which increases the online computation amount. In contrast, the proposed algorithm is robust to the incorrectly estimated SNR as much as the VSS-NSAF algorithm without any additional online computations because of its open-loop feature.

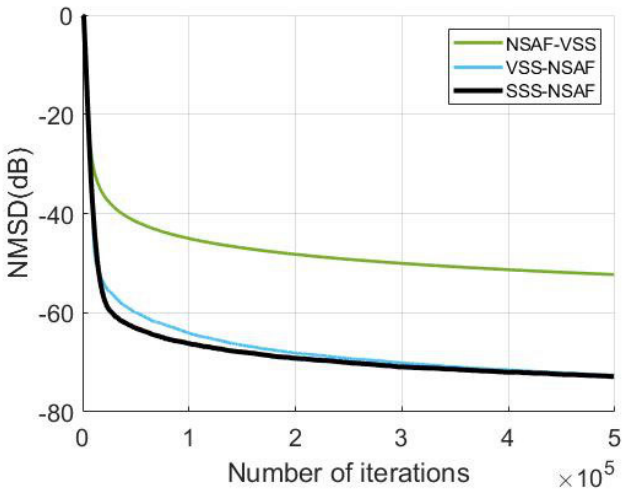

FIGURE 12. NMSD learning curves of the NSAF-VSS, VSS-NSAF and proposed algorithm in a misestimated SNR environment. The algorithms were all designed with 20dB SNR (M=1024, N=8, SNR=50dB, White Gaussian input).

VI. CONCLUSION

This article proposes an SS-NSAF algorithm to improve the convergence performance of NSAF without additional on-line computation. The MSD learning curve of the general NSAF algorithm was geometrically analyzed to create the objective curve. The step-size table was generated to force the MSD learning curve to follow the objective curve without any additional online computations. The generated step-size table was proportionally adjustable for the number of taps and the number of subbands. The novel memory efficient scheduling scheme minimizes the memory space required and at the same time greatly simplifies the operation. This brings very important advantages to chip level implementation. The simulation results showed that the proposed algorithm performs as well as the other variable step-size algorithms without any additional online computations and performs robustly in the wrong estimated SNR environment.

REFERENCES

[1] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Hoboken, NJ, USA: Wiley, 2003.
[2] S. S. Haykin, *Adaptive Filter Theory*. London, U.K.: Pearson, 2005.
[3] K.-A. Lee, W.-S. Gan, and S. M. Kuo, *Subband Adaptive Filtering: Theory and Implementation*. Hoboken, NJ, USA: Wiley, 2009.
[4] Y. V. Zakharov and J. Li, "Sliding-window homotopy adaptive filter for estimation of sparse UWA channels," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop (SAM)*, Jul. 2016, pp. 1–4.
[5] M. Yukawa, R. C. de Lamare, and R. Sampaio-Neto, "Efficient acoustic echo cancellation with reduced-rank adaptive filtering based on selective decimation and adaptive interpolation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 4, pp. 696–710, May 2008.
[6] H.-C. Huang and J. Lee, "A new variable step-size NLMS algorithm and its performance analysis," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 2055–2060, Apr. 2012.
[7] J. Hur, M. Lee, D. Kim, and P. Park, "A variable step-size robust saturation algorithm against impulsive noises," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 10, pp. 2279–2283, Oct. 2020.
[8] L. Shi, H. Zhao, X. Zeng, and Y. Yu, "Variable step-size widely linear complex-valued NLMS algorithm and its performance analysis," *Signal Process.*, vol. 165, pp. 1–6, Dec. 2019.

[9] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn., I, Commun.*, vol. 67, no. 5, pp. 19–27, May 1984.
[10] M. Rupp, "A family of adaptive filter algorithms with decorrelating properties," *IEEE Trans. Signal Process.*, vol. 46, no. 3, pp. 771–775, Mar. 1998.
[11] C. Paleologu, S. Ciochina, and J. Benesty, "An efficient proportionate affine projection algorithm for echo cancellation," *IEEE Signal Process. Lett.*, vol. 17, no. 2, pp. 165–168, Feb. 2010.
[12] L. Rey Vega, H. Rey, and J. Benesty, "A robust variable step-size affine projection algorithm," *Signal Process.*, vol. 90, no. 9, pp. 2806–2810, Sep. 2010.
[13] M. Tanaka, Y. Kaneda, S. Makino, and J. Kojima, "Fast projection algorithm and its step size control," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, May 1995, pp. 945–948.
[14] S. L. Gay, "The fast affine projection algorithm," in *Acoustic Signal Processing for Telecommunication*. Cham, Switzerland: Springer, 2000, pp. 23–45.
[15] M. Tanaka, "Reduction of computation for high-order projection algorithm," in *Proc. IEICE Fall Conf.*, 1993, pp. A–103.
[16] S. L. Gay, "A fast converging, low complexity adaptive filtering algorithm," in *Proc. IEEE Workshop Appl. Signal Process. to Audio Acoust.*, Oct. 1993, pp. 4–7.
[17] F. Yang, M. Wu, P. Ji, and J. Yang, "Low-complexity implementation of the improved multiband-structured subband adaptive filter algorithm," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5133–5148, Oct. 2015.
[18] Y. V. Zakharov, "Low-complexity implementation of the affine projection algorithm," *IEEE Signal Process. Lett.*, vol. 15, pp. 557–560, 2008.
[19] F. Yang, M. Wu, J. Yang, and Z. Kuang, "A fast exact filtering approach to a family of affine projection-type algorithms," *Signal Process.*, vol. 101, pp. 1–10, Aug. 2014.
[20] F. Yang and J. Yang, "A comparative survey of fast affine projection algorithms," *Digit. Signal Process.*, vol. 83, pp. 297–322, Dec. 2018.
[21] S. Sandeep Pradham and V. U. Reddy, "A new approach to subband adaptive filtering," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 655–664, Mar. 1999.
[22] K. A. Lee and W. S. Gan, "Inherent decorrelating and least perturbation properties of the normalized subband adaptive filter," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4475–4480, Nov. 2006.
[23] 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, Sep. 2004.
[24] V. Mathews and S. Cho, "Improved convergence analysis of stochastic gradient adaptive filters using the sign algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 4, pp. 450–454, Apr. 1987.
[25] J. Hur, I. Song, and P. Park, "A variable step-size normalized subband adaptive filter with a step-size scaler against impulsive measurement noise," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 64, no. 7, pp. 842–846, Jul. 2017.
[26] J. Ni and F. Li, "A variable step-size matrix normalized subband adaptive filter," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 6, pp. 1290–1299, Aug. 2010.
[27] J. Shin, N. Kong, and P. Park, "Normalised subband adaptive filter with variable step size," *Electron. Lett.*, vol. 48, no. 4, pp. 204–206, 2012.
[28] J. Jin Jeong, K. Koo, G. Tae Choi, and S. Woo Kim, "A variable step size for normalized subband adaptive filters," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 906–909, Dec. 2012.
[29] J.-H. Seo and P. Park, "Variable individual step-size subband adaptive filtering algorithm," *Electron. Lett.*, vol. 50, no. 3, pp. 177–178, Jan. 2014.
[30] Y. Yu, H. Zhao, and B. Chen, "A new normalized subband adaptive filter algorithm with individual variable step sizes," *Circuits, Syst., Signal Process.*, vol. 35, no. 4, pp. 1407–1418, Apr. 2016.
[31] P. Wen and J. Zhang, "A novel variable step-size normalized subband adaptive filter based on mixed error cost function," *Signal Process.*, vol. 138, pp. 48–52, Sep. 2017.
[32] M. S. E. Abadi and J. H. Husøy, "Selective partial update and set-membership subband adaptive filters," *Signal Process.*, vol. 88, no. 10, pp. 2463–2471, Oct. 2008.
[33] S.-E. Kim, Y.-S. Choi, M.-K. Song, and W.-J. Song, "A subband adaptive filtering algorithm employing dynamic selection of subband filters," *IEEE Signal Process. Lett.*, vol. 17, no. 3, pp. 245–248, Mar. 2010.
[34] M.-K. Song, S.-E. Kim, Y.-S. Choi, and W.-J. Song, "Selective normalized subband adaptive filter with subband extension," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 60, no. 2, pp. 101–105, Feb. 2013.
[35] P. Park, M. Chang, and N. Kong, "Scheduled-stepsize NLMS algorithm," *IEEE Signal Process. Lett.*, vol. 16, no. 12, pp. 1055–1058, Dec. 2009.
[36] C. H. Lee and P. Park, "Scheduled-Step-Size affine projection algorithm," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 9, pp. 2034–2043, Sep. 2012.

TAESU PARK (Student Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2015, and the M.S. degree in electrical engineering from the Pohang University of Science and Technology, Pohang, South Korea, in 2017, where he is currently pursuing the Ph.D. degree. His research interests include signal processing, state estimation, and artificial intelligence.

MINHO LEE (Graduate Student Member, IEEE) received the B.S. degree in electronics engineering from Kyungpook National University, Daegu, South Korea, in 2015, and the M.S. degree in electrical engineering from the Pohang University of Science and Technology, Pohang, South Korea, in 2018, where he is currently pursuing the Ph.D. degree. His research interests include signal processing, state estimation, and artificial intelligence.

POOGYEON PARK (Senior Member, IEEE) received the B.S. degree in control and instrumentation engineering, and the M.S. degree in control and instrumentation engineering from Seoul National University, Seoul, South Korea, in 1988 and 1990, respectively, and the Ph.D. degree from Stanford University, Stanford, CA, USA, in 1995.

From 1996 to 2000, he was an Assistant Professor with the Pohang University of Science and Technology, where he has been a Professor with the Electronic Electrical Engineering Department, since 2006. He has authored over 170 articles and the total citation for his articles is 9712. His research interest includes control and signal processing.