

Upgraded NLMS algorithm for speech enhancement with sparse and dispersive impulse responses

S N Sundaradhas* , S P Panchama moorthy and S S K Ramapackiyam

Department of Electronics and Communication Engineering, MepcoSchlenk Engineering College, Sivakasi, India

Received: 30 April 2019 / Accepted: 19 November 2019 / Published online: 10 February 2020

Abstract: To have the effective speech communication, the information should be clearly passed in a noise-free environment. However, in real-world environment, the existence of background noise degrades the performance of the system. Based on the blind source separation strategy, various adaptive algorithms are designed and implemented using both dispersive impulse response and sparse impulse response. Even though the existing dual fast normalized least mean square algorithm works well under different noisy situations and gives a good performance, the problem is that it involves large number of processing steps. To overcome the complexity in finding the signal prediction parameter and to improve the performance of speech enhancement, we propose three adaptive filtering algorithms namely revised twofold rapid normalized least mean square algorithm, diminished twofold normalized least mean square algorithm and upgraded balanced twofold normalized least mean square algorithm (UBTNLMS). Taking the performance objective criteria into account, these algorithms have been tested for segmental signal-to-noise ratio, segmental mean square error, signal-to-noise ratio, mean square error and cepstral distance. On comparing the performance of the existing and proposed algorithms, UBTNLMS performs better than the other algorithms.

Keywords: Biophysical techniques; General linear acoustics; Remote sensing; LIDAR and adaptive systems; Acoustics signal processing

PACS Nos.: 87.80.-y; 43.20.+g; 42.68.Wt; 43.60.-c

1. Introduction

Human being naturally possesses variety of ways to get back the information from the exterior world and much aware how to communicate with each other. Speech, images and text are the three most essential sources of information. For most of the cases, speech stands as the most resourceful and comfortable one. Speech not only delivers the content of linguistic information, but also delivers the extra valuable information like the mood of the speaker. Once the speaker and listener are close to each other in a calm situation, communication is normally easy and accurate. On the other hand, when the listener and speaker are at a long distance or in a noisy background situation, the listener's ability to understand the particular information may get diminished. So, in most of the speech

communication systems, the quality and intelligibility of speech play a major role. To reduce the noise and to improve the speech quality, several enhancement algorithms have been developed over past decades [1, 2]. Adaptive filters [3] are very important in telecommunication systems such as hand-free telephony [4], teleconferencing systems [5], hearing aids [6] with acoustic noise cancellation [7] and speech quality enhancement.

The most accepted single channel speech enhancement algorithms include minimum mean square error estimation (MMSE) [8], spectral subtraction [9] and log minimum mean square error (logMMSE) [10]. Some of these techniques have limitations in speech enhancement applications [11]. The alternate approach to speech enhancement is the signal subspace method [12]; the signal subspace method is based on the singular value decomposition [13]. In time domain and frequency domain, several algorithms have been developed [14–16], the double affine projection algorithm [17], double fast Newton transversal filter [18] and double pseudo double affine projection algorithm [19],

*Corresponding author, E-mail: nidhyan@mepcoeng.ac.in

partial update double affine projection algorithm [20], and sub-band algorithm [7]. In a moving car environment, a recent technique named blind source separation is used [21]. In the literature, we find two widely used structures of blind source separation, the forward blind source separation [22] and the backward blind source separation [23]. Forward blind source algorithm is used to update the cross-filters using recursive least square algorithm [24]. The symmetric adaptive de correlation (SAD) algorithms of forward and backward types [25–27] are essential keys to detach the speech signal from noisy observations. A dual fast normalized least mean square algorithm is used to improve the intelligibility of the speech enhancement systems [18, 28]. In most of the cases, blind source separation algorithms are used to find the dispersive impulse response. But this paper concentrates on dispersive and sparse impulse responses of convolution mixture. In this paper, we have revised and discussed three different types of dual channel adaptive algorithms for speech enhancement. This paper is organized as follows: in Sect. 2, we discuss about the convolution mixture model of the adaptive filters. In Sect. 3, we discuss about the mathematical formulation for the existing DFNLMS algorithm. In Sect. 4, we discuss about the proposed methods. In Sect. 5, the comparison results of the various developed algorithms are given, and finally, the conclusion is given in Sect. 6.

2. Convolution mixture model

Convolution is a mathematical operation on two functions to produce a third function that expresses how the shape of one is modified by the other. Convolution is used here to mix speech and noise signals. The convolution mixture model consists of two components: an original speech signal and a noise signal. The original speech signal $a(n)$ and noise signal $b(n)$ are convolutedly mixed with two impulse responses, i.e. dispersive and sparse impulse responses of mixture contain $(h_{11}(n), h_{22}(n), h_{12}(n)$ and $h_{21}(n))$ as shown in Fig. 1. The output signal of the convoluted mixture model is given as below

$$s_1(n) = a(n) * h_{11}(n) + b(n) * h_{21}(n) \quad (1)$$

$$s_2(n) = b(n) * h_{22}(n) + a(n) * h_{12}(n) \quad (2)$$

where (*) represents the convolution operation. Consider the two direct acoustic paths through $h_{11}(n)$ and $h_{22}(n)$ are assumed to be identity, i.e. $h_{11}(n) = h_{22}(n) = \delta(n)$. Hence, here we consider the dispersive and sparse impulse responses between the cross-coupling effects of the channels $h_{12}(n)$ and $h_{21}(n)$. The two acoustic paths are found to be unity, and hence, the model is further simplified and is given by the following expression

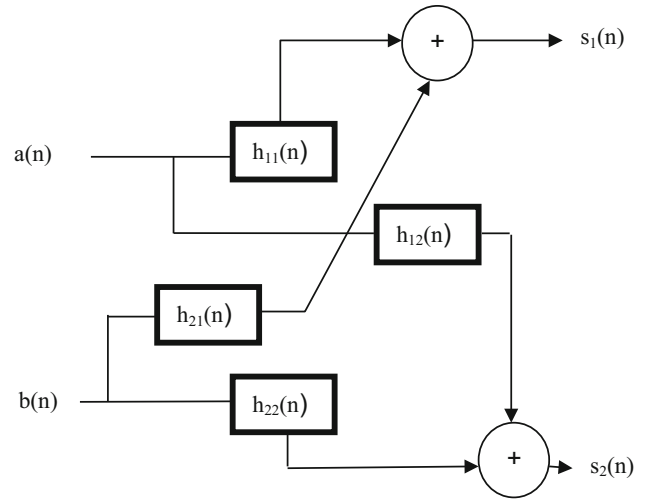


Fig. 1 Convolution mixture model

$$k_1(n) = a(n) + b(n) * h_{21}(n) \quad (3)$$

$$k_2(n) = b(n) + a(n) * h_{12}(n). \quad (4)$$

Sparse impulse response is nothing but an impulse response that consists only a small percentage of its components with a significant magnitude, while others are zeros which is shown in Fig. 2(a) for the adaptive filter length $L = 64$. The vice versa is the dispersive impulse response, i.e. large number of components having its significant magnitude and small portions are containing zeros which is shown in Fig. 2(b).

3. Existing method

In this section, we discuss about the mathematical formulation of the dual fast normalized least mean square method [20]. This algorithm is used to update the two cross-filters. To estimate the original speech signal, the adaptive filters $p_{21}(n)$ and $p_{12}(n)$ are used. From the mixing model, the enhanced speech output is given below:

$$s_1(n) = a(n) - p_1^T(n)K_2(n) \quad (5)$$

$$s_2(n) = b(n) - p_2^T(n)K_1(n) \quad (6)$$

where $K_1(n) = [k_1(n), k_1(n-1), \dots, k_1(n-L+1)]^T$ and $K_2(n) = [k_2(n), k_2(n-1), \dots, k_2(n-L+1)]^T$.

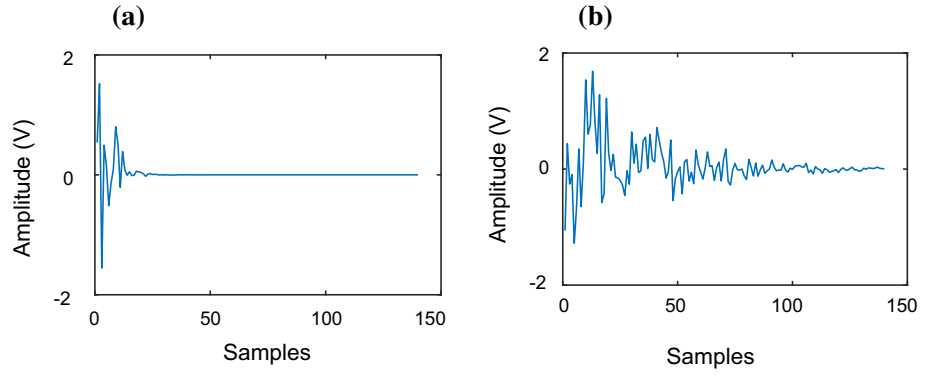
The adaptive filter is used for updating the relation which is given as:

$$p_1(n+1) = p_1(n) - \mu_1(s_1(n)d_1(n)) \quad (7)$$

$$p_2(n+1) = p_2(n) - \mu_2(s_2(n)d_2(n)) \quad (8)$$

where $0 < \mu_1, \mu_2 < 2$ are the two step sizes which control the convergence behaviour of the adaptive filters. The

Fig. 2 Impulse response with $L = 64$. (a) Sparse impulse response (b) dispersive impulse response



adaptation gains $d_1(n)$ and $d_2(n)$ are given by the following relation:

$$d_1(n) = \gamma_1(n)N_1(n) \quad (9)$$

$$d_2(n) = \gamma_2(n)N_2(n) \quad (10)$$

where $\gamma_1(n)$, $\gamma_2(n)$ and $N_1(n)$, $N_2(n)$ are the likelihood variables and dual Kalman gain, respectively. The likelihood variables $\gamma_1(n)$ and $\gamma_2(n)$ and Kalman variables $N_1(n)$, $N_2(n)$ can be calculated as follows:

$$\gamma_1(n) = \frac{1}{1 - N_1^T K_2(n)} \quad (11)$$

$$\gamma_2(n) = \frac{1}{1 - N_2^T K_1(n)} \quad (12)$$

$$\begin{bmatrix} N_1(n) \\ * \end{bmatrix} = \frac{c_1(n)}{\lambda Z_1(n-1) + c_0} \begin{bmatrix} -1 \\ N_1(n-1) \frac{\lambda Z_1(n-1) + c_0}{c_1(n)} \end{bmatrix} \quad (13)$$

$$\begin{bmatrix} N_2(n) \\ * \end{bmatrix} = \frac{c_2(n)}{\lambda Z_2(n-1) + c_0} \begin{bmatrix} -1 \\ N_2(n-1) \frac{\lambda Z_2(n-1) + c_0}{c_2(n)} \end{bmatrix} \quad (14)$$

where (*) represents the last element of the Kalman variable, c_0 is a small positive constant, λ ($0 < \lambda < 1$) is an exponential forgetting factor, and λ_a is a small positive constant.

The parameters $Z_1(n)$ and $Z_2(n)$ are the forward prediction errors variances, and $c_1(n)$ and $c_2(n)$ are given below:

$$Z_1(n) = \lambda_a Z_1(n-1) + c_1^2(n) \quad (15)$$

$$Z_2(n) = \lambda_a Z_2(n-1) + c_2^2(n) \quad (16)$$

$$c_1(n) = k_2(n) - g_1 k_2(n-1) \quad (17)$$

$$c_2(n) = k_1(n) - g_2 k_1(n-1). \quad (18)$$

The variables $g_1(n)$ and $g_2(n)$ can be calculated as

$$g_1(n) = \frac{f_1(n)}{f_2(n) + c_a} \quad (19)$$

$$g_2(n) = \frac{f_3(n)}{f_4(n) + c_a} \quad (20)$$

where $f_1(n)$ and $f_2(n)$ represent the first autocorrelation mixture of $k_2(n)$, and $f_3(n)$ and $f_4(n)$ represent the first autocorrelation mixture of $k_2(n)$. The following relations $f_1(n)$, $f_2(n)$, $f_3(n)$ and $f_4(n)$ are estimated recursively by

$$f_1(n) = \lambda_a f_1(n-1) + k_2(n)k_2(n-1) \quad (21)$$

$$f_2(n) = \lambda_a f_2(n-1) + k_2^2(n) \quad (22)$$

$$f_3(n) = \lambda_a f_3(n-1) + k_1(n)k_2(n-1) \quad (23)$$

$$f_4(n) = \lambda_a f_4(n-1) + k_1^2(n). \quad (24)$$

4. Proposed algorithms

Under different noisy circumstances, the existing DFNLM algorithm works well and offers good performance. But the challenge lies in estimating the parameters of the signal prediction using vast computation. In order to eliminate the problem of signal prediction parameter and to improve speech enhancement performance, we proposed three algorithms derived as follows:

4.1. Revised twofold rapid normalized least mean square (RTRNLMS) algorithm

The main objective of this RTRNLMS algorithm is to reduce the steps of derivation in the existing DFNLM algorithm and to overcome the complexity of signal prediction problem. In the RTRNLMS method, we neglect the parameters, $g_1(n)$ and $g_2(n)$ from Eqs. (17) and (18), respectively, because the signal prediction parameter has little significance and offered no such greater improvement in performance of the speech. In RTRNLMS, we get a better performance by subtracting the previous values

directly. The mathematical expression for the RTRNLMS is given below:

The parameter error variances are as in (15) and (16). The $c_1(n)$ and $c_2(n)$ are given as:

$$c_1(n) = k_2(n) - k_2(n-1) \quad (25)$$

$$c_2(n) = k_1(n) - k_1(n-1). \quad (26)$$

In (25), by subtracting the previous value of mixture k_2 from the present value of mixture k_2 , we get a better improvement in the speech quality. The mixture k_1 is obtained by convolving the impulse response with noise signal and then added with the speech signal, and mixture k_2 is obtained by convolving the impulse response with speech signal and then added with noise signal. The likelihood variables $\gamma_1(n)$ and $\gamma_2(n)$ are given in (11) and (12). The Kalman variables $N_1(n)$, $N_2(n)$ can be calculated as follows:

$$\begin{bmatrix} N_1(n) \\ * \end{bmatrix} = -\frac{c_1(n)}{\lambda Z_1(n-1) + c_0} + N_1(n-1) \quad (27)$$

$$\begin{bmatrix} N_2(n) \\ * \end{bmatrix} = -\frac{c_2(n)}{\lambda Z_2(n-1) + c_0} + N_2(n-1). \quad (28)$$

The adaptation gains $d_1(n)$ and $d_2(n)$ are given in (9) and (10), the updating is as per (7) and (8), and the enhanced speech $s_1(n)$ and $s_2(n)$ are as given in (5) and (6).

4.2. Diminished twofold normalized least mean square (DTNLMS) algorithm

In diminished twofold normalized least mean square method, the derivation steps are minimized to few steps. The filter weight update equation used in DTNLMS is different from the existing and the RTRNLMS algorithms. In DTNLMS, we used the enhanced speech convolved with the segments of enhanced speech, instead of mixtures of noise that we used in DFNLMS and RTRNLMS. By using the weight adaptation method, we get a better improvement in the enhanced speech. The controlled parameter used here is with the step size $\mu_1 = \mu_2 = 0.1$, and $\beta = 10^{-2}$

The estimated speech signal is given by,

$$s_1(n) = a(n) - u_1^T(n)K_2(n) \quad (29)$$

$$s_2(n) = b(n) - u_2^T(n)K_1(n). \quad (30)$$

For updating the adaptive filter, the expression is given by,

$$p_1(n+1) = p_1(n) - (\mu_1(s_1(n)s_2(n))/\varepsilon) \quad (31)$$

$$p_2(n+1) = p_2(n) - (\mu_2(s_2(n)s_1(n))/\varepsilon) \quad (32)$$

where $s_1(n) = [s_1(n), s_1(n-1), \dots, s_1(n-L+1)]^T$, and $s_2(n) = [s_2(n), s_2(n-1), \dots, s_2(n-L+1)]^T$, ε is a small positive constant. The $K_1(n)$ and $K_2(n)$ are defined in

Sect. 3. The filter weight adaptation is obtained by convolving the enhanced speech signal with the segments of enhanced speech signal and also with the step size and then divided by the controlled parameter and then it is subtracted from the value of the weight in adaptive filter. The enhanced speech signal depends on the β value. By properly selecting the step sizes and β , we get the better enhanced speech quality.

4.3. Upgraded balanced twofold normalized least mean square (UBTNLMS) algorithm

DTNLMS gives a better result when compared to DFNLMS and RTRNLMS. There exist few issues to enhance the speech signal, because it is controlled by a single variable β . We cannot rely on adjusting that single value to obtain the enhanced speech signal. So, we go for the UBTNLMS method. In UBTNLMS method, we used many parameters to obtain the enhanced speech. The controlled parameter used here is with step size $\mu_1 = \mu_2 = 0.1$, and $\xi = 1$ and α is -0.5 . In UBTNLMS, we used a different updating expression for the adaptive filter and the adaptive filter is designed with diagonal weight elements. Diagonal matrix contains the value only in the diagonals, remaining are zeros. This algorithm possesses the stability of the outcomes with the best performance by the balanced control parameters. So this algorithm is named as upgraded balanced twofold normalized least mean square algorithm. The two diagonal step size control matrixes $w_{12}(n)$ and $w_{21}(n)$ are given by,

$$\begin{aligned} w_{12}(n) &= \text{diag}\{w_{12,0}(n), w_{12,1}(n), \dots, w_{12,L-1}(n-1)\} \\ &= \begin{bmatrix} w_{12,0}(n) & 0 & \cdots & 0 \\ 0 & w_{12,1}(n) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_{12,L-1}(n) \end{bmatrix} \\ w_{21}(n) &= \text{diag}\{w_{21,0}(n), w_{21,1}(n), \dots, w_{21,L-1}(n-1)\} \\ &= \begin{bmatrix} w_{21,0}(n) & 0 & \cdots & 0 \\ 0 & w_{21,1}(n) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_{21,L-1}(n) \end{bmatrix} \\ w_{12,i}(n) &= \frac{1-\alpha}{2L} + (1+\alpha) \frac{1}{2 \times \left(\sum_{i=0}^{L-1} w_{12,i}(n) \right) + \varphi} \end{aligned} \quad (33)$$

$$w_{21,l}(n) = \frac{1 - \alpha}{2L} + (1 + \alpha) \frac{1}{2 \times \left(\sum_{i=0}^{L-1} w_{21,i}(n) \right) + \varphi}. \quad (34)$$

The α is a small value takes between -1 and 1 , and φ is a small positive constant. In UBTNLMS, we have used a different updating formulas for the adaptive filter. We have used diagonal weight elements of the diagonal step size control matrixes $w_{12}(n)$ and $w_{21}(n)$ in the adaptive filter Eqs. (33) and (34). By using this step in UBTNLMS algorithm, it possesses stability of the outcomes and better performance with the support of balanced control parameters.

The tap weight adaptation is given by,

$$p_{12}(n) = p_{12}(n-1) + \mu_1 \frac{w_{12}(n-1)s_1(n)s_2(n)}{K_1^T(n)K_1(n) + \xi} \quad (35)$$

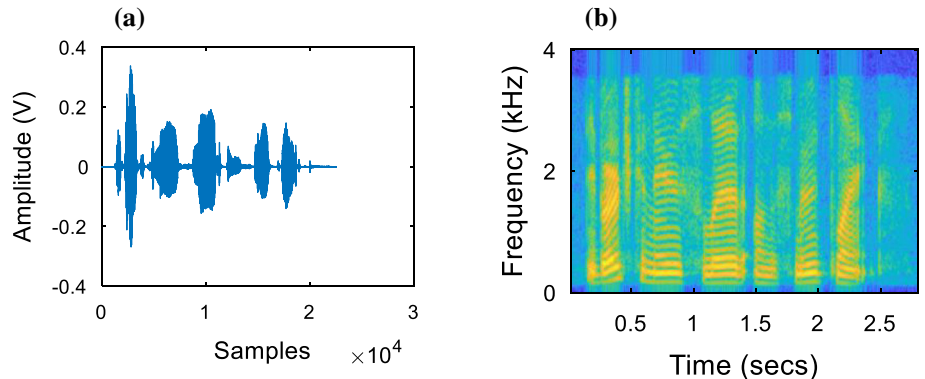
$$p_{21}(n) = p_{21}(n-1) + \mu_2 \frac{w_{21}(n-1)s_1(n)s_2(n)}{K_2^T(n)K_2(n) + \xi}. \quad (36)$$

The weight adaptation is obtained by multiplying the previous values of diagonal elements with the enhanced speech signal along with segments of enhanced speech signal and then divided with the sum of mixtures of k_1 and ξ . The value of ξ is 1 , and the value of α is -0.5 . By substituting the tap weight adaptation in the output, we get the enhanced speech as per (5) and (6). Based on the update equations of the two adaptive filters (35) and (36) and the diagonal elements presented in $w_{21}(n)$ and $w_{12}(n)$, enhanced speech signal is obtained.

5. Evaluation of simulation result

In this section, evaluation results of the proposed algorithms are discussed. From Sect. 2, we take the convolution mixture model for mixing the noise and the speech. Figure 3 shows a clean speech and its spectrogram, while Fig. 4 represents a noise signal and its spectrogram.

Fig. 3 (a) Clean speech signal and (b) spectrogram of clean speech signal



The original speech signal and noise signal are subjected to convolution operation with two impulse responses, i.e. dispersive and sparse impulse responses. An impulse response is sparse, if a large fraction of its energy is concentrated in a small fraction of its duration. According to [18], sparse and dispersive impulse response sequences are generated by using the expression $h(n) = Ae^{-Bn}$, where A represents a scalar factor that is equal to one and B is a damping factor which is related to the reverberation time tr . The damping factor for dispersive impulse response is $B = 3\log(10)/tr$. Therefore, the decay factor is 10^{-3} . For the sparse impulse response, the damping factor is $B = 20\log(10)/tr$. Therefore, the decay factor is 10^{-20} which is a very less compared to 10^{-3} . Hence, in dispersive impulse response, the damping decays very slowly and has more damping than sparse. Sparse impulse response decays fast and has more zero values. The impulse response is generated when the length of the adaptive filter is taken as $L = 64$

The database used here is NOIZEOUS database with the sampling rate = 8 kHz. It consists of 30 speakers that contain different noises like car, train, airport, street, station, restaurant noise at input with SNR of 0, 5, 10 and 15 dB.

Figure 5 represents the mixing samples of mixture 1 and 2. Figure 6 represents the enhanced speech signal. In this section, we compare the performance of the existing algorithm with the proposed algorithms. The first proposed algorithm is revised twofold normalized least mean square (RTRNLMS) algorithm. This algorithm improves the performance by avoiding the problem of prediction parameters. The second proposed algorithm is the diminished twofold normalized least mean square (DTNLMS) algorithm. This algorithm reduces the level of noise to a maximum level. The third proposed algorithm is upgraded balanced twofold normalized least mean square algorithm (UBTNLMS). This algorithm used a diagonal matrix in the weight update equation to improve the performance of enhanced speech signal. The parameters of the existing and proposed algorithm are summarized in Table 1.

Fig. 4 (a) Noise signal and (b) spectrogram of noise signal

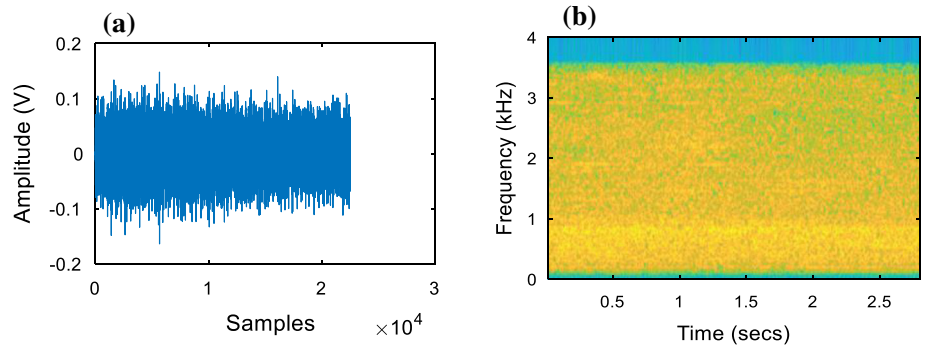


Fig. 5 (a) Mixing samples of mixture 1 and (b) mixing samples of mixture 2

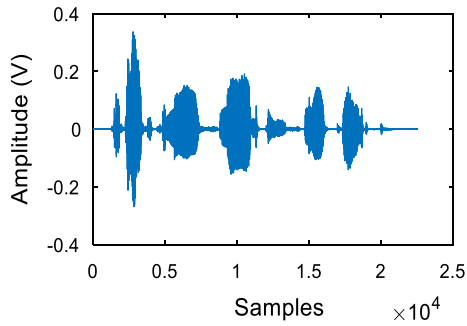
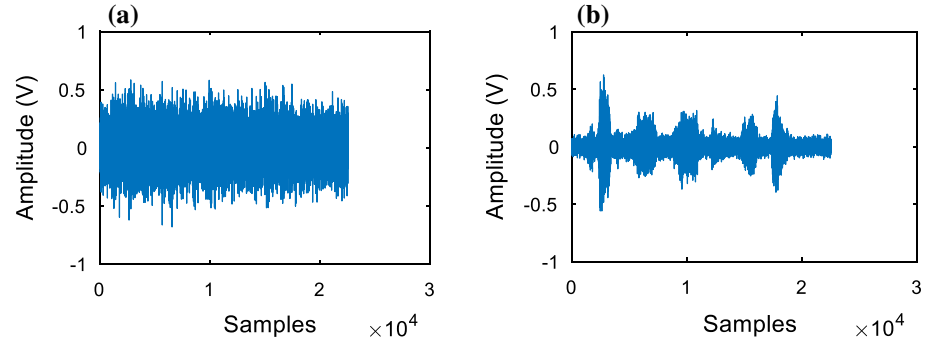


Fig. 6 Enhanced speech signal

In the case of DFNLMS algorithm, we have selected the optimum step size as $\mu_1 = \mu_2 = 0.1$, forgetting factor $\lambda = 0.09$, $\lambda_a = 0.1$ and constant $c_0 = 0.0009$, $c_a = 1$. However, in RTRNLMS algorithm, we have fixed the step size as $\mu_1 = \mu_2 = 0.1$, forgetting factor $\lambda = 0.09$, $\lambda_a = 0.1$ and constant $c_0 = 0.0009$. Similarly, for DTNLMS and UBTNLMS, we fixed the step size as $\mu_1 = \mu_2 = 0.1$, $\varepsilon = 1$, $\alpha = -0.5$ and $\varphi = 10^{-2}$. The values for the parameters are selected by referring the parameter ranges from the literature [21], and then by iteratively fixing the parameter values, it gives the best result by implementing the algorithm. In RTRNLMS algorithm, λ varies between $0 < \lambda < 1$ and similarly step size μ also changes from 0 to 1. So the values are randomly selected between 0 to 1 for both μ and λ . For different values of $\mu_1 = \mu_2$ such as 0.8, 0.6, 0.4, 0.2, 0.1, the corresponding readings are taken for $\lambda = 0.09$, 0.06, 0.04, 0.01. From this, we infer that for 0 dB, 5 dB,

10 dB and 15 dB inputs, the parameters $\mu_1 = \mu_2 = 0.1$ and $\lambda = 0.09$ gave a comparatively good improvement and these parameter values are used for the enhancement in RTRNLMS algorithm. Similarly, in DTNLMS and UBTNLMS algorithms, the same iterative procedure is used for fixing the parameter values.

The size of the adaptive filter used in this work is 64. For simulation results, we have taken the filter size as 64, 128 and 256. The filter weights are adjusted according to the filter weight update equation of the respective algorithm, until it gives the best result. In UBTNLMS, we used a different updating formulas for the adaptive filter. The balanced control parameters in this algorithm ensure performance stability. When compared to other algorithms, UBTNLMS performs best in enhancing the speech. To analyse the quality of the speech signal, in the next section, we present the comparative results of the existing and proposed algorithms in terms of the objective measures (1) the segmental signal-to-noise ratio, (2) the segmental mean square error, (3) signal-to-noise ratio, (4) cepstral distance and (5) mean square error.

5.1. Segmental signal-to-noise ratio (segSNR)

It is defined as the ratio of the enhanced speech signal to that of the background noise. The segmental signal-to-noise ratio is calculated by the following relation:

Table 1 Simulation parameter values of the existing and proposed algorithm with $L = 64$

| Algorithms | Existing DFNLMS | Proposed algorithms | | |
|-----------------------------|---|--|--|--|
| | | RTR NLMS | DT NLMS | UBT NLMS |
| Controlled parameter values | $\mu_1 = \mu_2 = 0.1$ $\lambda = 0.09$ $\lambda_a = 0.1$ $c_0 = 0.0009$ $c_a = 1$ | $\mu_1 = \mu_2 = 0.1$ $\lambda = 0.09$ $\lambda_a = 0.1$ $c_0 = 0.0009$ | $\mu_1 = \mu_2 = 0.1$ $\beta = 10^{-2}$ | $\mu_1 = \mu_2 = 0.1$ $\alpha = -0.5$ $\varphi = 10^{-2}$ $\xi = 1$ |

$$(\text{segsnr})_{\text{dB}} = 10 \log_{10} \left(\frac{\sum_{i=0}^R |a(i)|^2}{\sum_{i=0}^R |a(i) - S_1(i)|^2} \right) \quad (37)$$

where R is the number of frames and s_1 represents the enhanced speech signal, and a is the clean speech signal. The VAD is the voice activity detector which detects the presence and absence of speech and noise components. The segmental signal-to-noise ratio is calculated only in presence of speech only presence periods. In segmental SNR, the block size represents the total number of frame segments which has voice activity. After VAD detection, if the length of the speech only presence region has 9000 samples for the given input speech signal, we split the region into 100×90 frames. The total frame is now 100, and each frame consists of 90 samples. The 100 frames represent the block size. The block size depends on frame size. First, we divide the presence of speech regions into frames, where each frame consists of segments of particular length. In this paper, we have split the speech only presence regions into 45 frames and each frame consists of 75 samples.

Figures 7, 8, 9 and 10 portray the measure segmental SNR for both dispersive impulse and sparse responses for $L = 64$ and 128. The impulse responses at 0 dB and 5 dB are analysed for the bloc of 45 samples. Among the existing and proposed algorithms, the UBTNLMS algorithm responded with the better response. The SegSNR is about 98 dB at 0 dB and 100 dB at 5 dB. The sparse responses are also analysed at 0 dB and 5 for a bloc of 45

samples which give the performance better than the dispersive impulse response. At 0 dB, the SegSNR is about 100 dB which is higher than the dispersive impulse responses. By increasing the adaptive length of the filter, sparse impulse response performs better when compared to dispersive impulse response.

5.2. Segmental mean square error (SegMSE)

It is defined as the difference between the square of the true values and the estimated value. The segmental mean square error is given by the following relation

$$(\text{segmse})_{\text{dB}} = \left(10 \log_{10} \sum_{i=0}^R |a(i) - S_1(i)|^2 \right) \quad (38)$$

where R is the number of frames and s_1 represents the enhanced speech signal, and a is the clean speech signal. The segmental mean square error ratio is calculated only in the absence of speech only presence periods, i.e. only in noise presence periods. First, we divide the noise only presence regions into frames and each frame consists of segments of particular length. In this paper, we have split the speech signal into 55 frames and each frame with of 75 samples.

Figures 7, 8, 9, 10, 11, 12, 13 and 14 portray the performance measure with segmental SNR and segmental MSE for both dispersive impulse and sparse impulse for length $L = 64$ and 128. The impulse responses at 0 dB and

Fig. 7 SegSNR for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) and by using the dispersive impulse response. The adaptive filter length is taken as $L = 64$

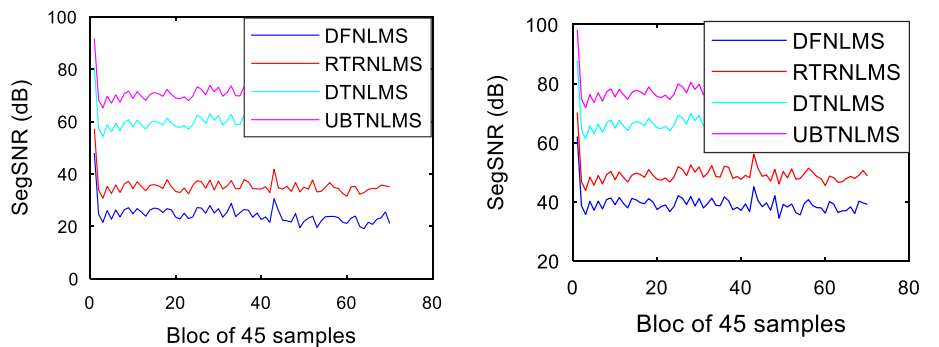


Fig. 8 SegSNR for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (Right) by using the sparse impulse response. The adaptive filter length is taken as $L = 64$

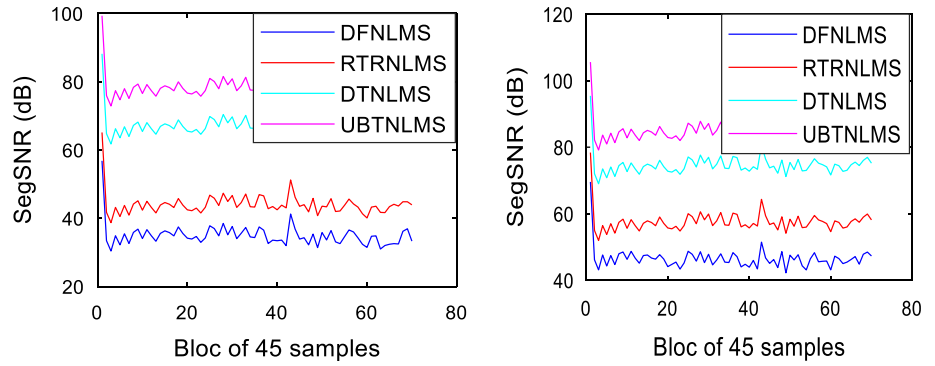


Fig. 9 SegSNR for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the dispersive impulse response. The adaptive filter length is taken as $L = 128$

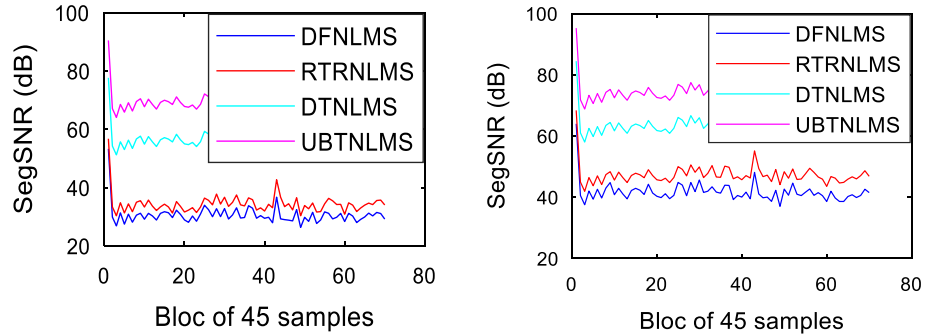


Fig. 10 SegSNR for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the sparse impulse response. The adaptive filter length is taken as $L = 128$

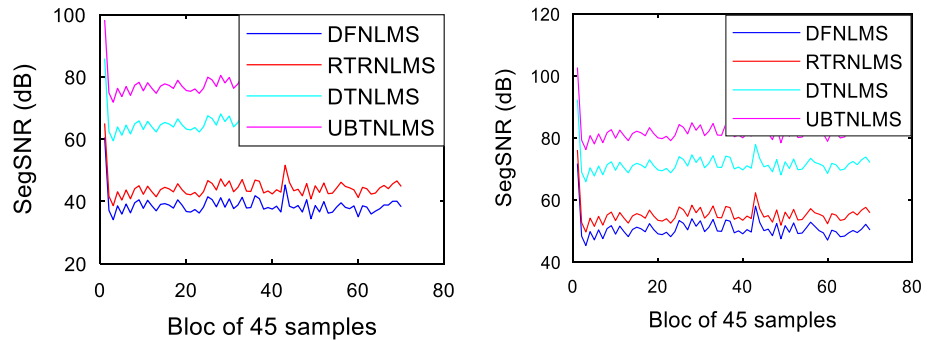
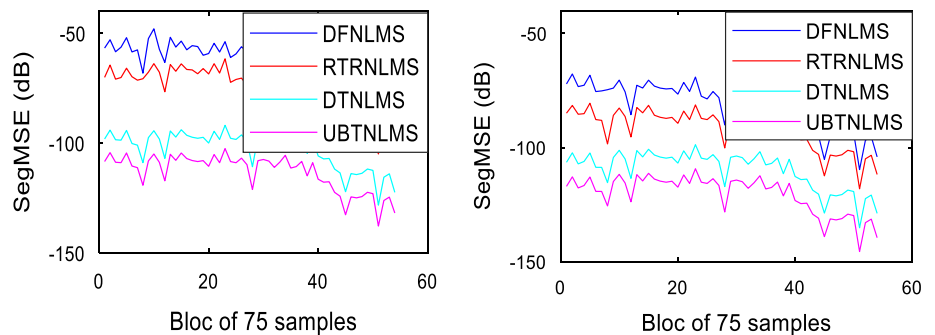


Fig. 11 SegMSE for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the dispersive impulse response. The adaptive filter length is taken as $L = 64$



5 dB are analysed for the bloc of 45 samples for segmental SNR, and the bloc of 75 samples is analysed for segmental MSE. Among the discussed algorithms, the UBTNLMS algorithm responded with the better response in both dispersive and spare impulse responses. From Figs. 5, 6, 7 and

8, it is clear that the sparse impulse response performs better than the dispersive impulse response. By increasing the adaptive filter length, sparse impulse response performs better than dispersive impulse response.

Fig. 12 SegMSE for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the dispersive impulse response. The adaptive filter length is taken as $L = 64$

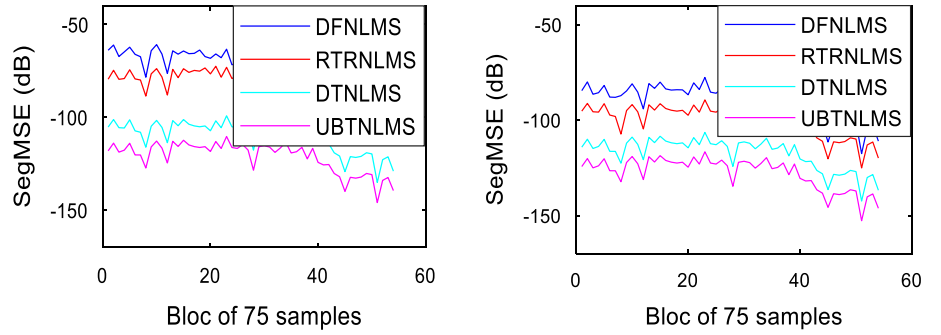


Fig. 13 SegMSE for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the dispersive impulse response. The adaptive filter length is taken as $L = 128$

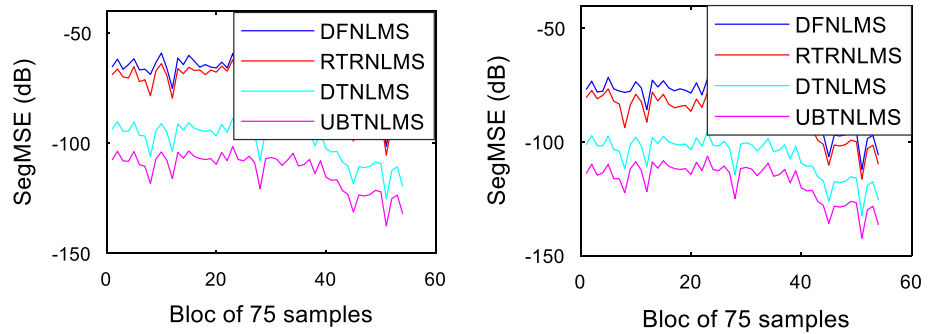
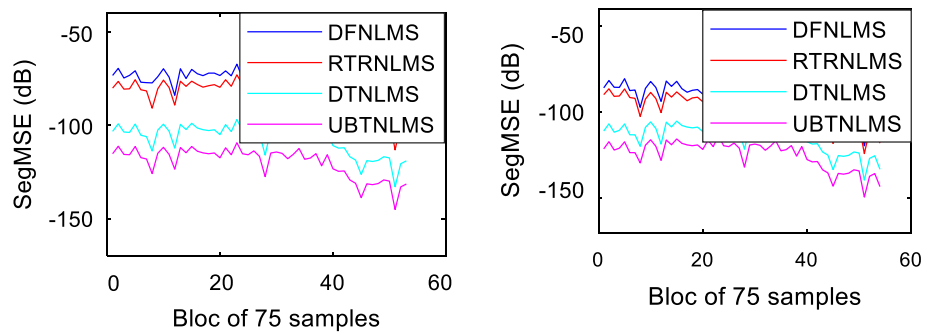


Fig. 14 SegMSE for the proposed and existing algorithm having the input SNR 0 dB (left) and 5 dB (right) by using the sparse impulse response. The adaptive filter length is taken as $L = 128$



5.3. Signal-to-noise ratio (SNR)

It is defined as the ratio of the speech signal to that of noise signal. The signal-to-noise ratio is given in (37). Here, SNR is calculated for total length of the samples.

5.4. Cepstral distance (CD)

It is used to measure the distortion present in the signal. The cepstral distance is given by the relation,

$$CD_{dB} = \sum_{i=1}^R \text{ISFT}[\log(|a(\omega, i)|) - \log(|s_1(\omega, i)|)]^2 \quad (39)$$

where ISFT denotes the inverse short-time Fourier transform. The STFT of the original speech signal and enhanced speech signal is $a(\omega, i)$, and $s_1(\omega, i)$, respectively.

5.5. Mean square error (MSE)

Mean square error is calculated for the difference between the true values and the estimated value. The mean square error equation is given in (38). Here, MSE is calculated for total length of the samples. Table 4 shows the performance of MSE for the existing and proposed algorithms for dispersive and sparse impulse responses. From Table 4, we infer that, at 0 dB input SNR for dispersive impulse response for DFNLMS, the MSE is -65.90 dB, RTRNLMS has -73.17 dB, DTNLMS has -98.34 dB, UBTNLMS has -98.55 dB and for sparse impulse response the existing DFNLMS has the MSE of -73.58 dB, RTRNLMS has -80.94 dB, DTNLMS has -106.24 dB, and UBTNLMS has -116.99 dB. On comparing, sparse impulse response shows better performance for all the proposed algorithms when compared to the dispersive impulse response.

Table 2 Performance of SNR for the existing and proposed algorithms in terms of dispersive and sparse impulse responses

| Noise type | Input SNR (dB) | Signal-to-noise ratio (SNR) in dB | | | | | | | |
|------------|----------------|-----------------------------------|---------------------|---------|----------|-------------------------|---------------------|---------|--------------|
| | | Dispersive impulse response | | | | Sparse impulse response | | | |
| | | Existing DFNLMS | Proposed algorithms | | | Existing DFNLMS | Proposed algorithms | | |
| | | | RTR NLMS | DT NLMS | UBT NLMS | | RTR NLMS | DT NLMS | UBT NLMS |
| Station | 0 | 19.57 | 28.23 | 42.85 | 63.74 | 27.58 | 35.33 | 42.66 | 71.79 |
| | 5 | 28.19 | 38.20 | 49.66 | 64.24 | 35.45 | 46.22 | 48.88 | 72.72 |
| | 10 | 38.58 | 47.83 | 50.46 | 71.93 | 47.29 | 55.16 | 60.90 | 82.50 |
| | 15 | 50.39 | 56.56 | 58.54 | 76.11 | 57.37 | 58.92 | 64.44 | 83.42 |
| Street | 0 | 18.84 | 26.23 | 42.49 | 61.00 | 26.53 | 34.16 | 42.76 | 69.33 |
| | 5 | 26.90 | 34.28 | 47.34 | 68.66 | 34.52 | 42.68 | 47.55 | 76.63 |
| | 10 | 40.96 | 47.51 | 54.93 | 77.83 | 49.67 | 55.13 | 57.77 | 84.34 |
| | 15 | 51.01 | 59.67 | 59.74 | 75.89 | 57.37 | 58.92 | 64.44 | 83.42 |
| Car | 0 | 18.34 | 26.25 | 42.50 | 60.36 | 25.20 | 34.14 | 43.04 | 68.51 |
| | 5 | 30.38 | 38.50 | 49.43 | 67.18 | 38.09 | 46.15 | 49.53 | 74.81 |
| | 10 | 40.38 | 52.53 | 54.86 | 71.96 | 46.22 | 53.98 | 58.46 | 80.56 |
| | 15 | 49.46 | 59.16 | 59.95 | 72.95 | 57.65 | 59.82 | 67.87 | 80.98 |
| Babble | 0 | 23.68 | 27.72 | 44.83 | 62.84 | 31.59 | 35.85 | 44.90 | 70.73 |
| | 5 | 31.33 | 37.71 | 50.01 | 70.96 | 42.51 | 47.21 | 51.93 | 78.93 |
| | 10 | 39.82 | 44.21 | 51.96 | 74.12 | 48.27 | 52.46 | 52.83 | 80.94 |
| | 15 | 47.39 | 56.78 | 57.82 | 74.21 | 57.65 | 59.82 | 67.87 | 81.98 |

The bold in the last column of Table is the indication and quantitative evidence of superior performance of UBT NLMS algorithm over RTR NLMS and DT NLMS algorithms in speech enhancement

Table 2 shows the SNR performance for the existing and proposed algorithms for dispersive and sparse impulse responses. From Table 2, we infer that, at 0 dB input SNR for dispersive impulse response for DFNLMS, the SNR is 19.57 dB, RTRNLMS has 28.23 dB, DTNLMS has 42.85 dB, UBTNLMS has 63.74 dB and for sparse impulse response the existing DFNLMS has the SNR of 27.58 dB, RTRNLMS has 35.33 dB, DTNLMS has 42.66 dB, and UBTNLMS has 71.79 dB. On comparing with dispersive and sparse impulse responses, sparse impulse response shows better performance for all the algorithms. Table 3 shows the performance of CD for the existing and proposed algorithms for dispersive and sparse impulse responses. From Table 3, we infer that, at 0 dB for dispersive impulse response for DFNLMS, the cepstral distance is -2.49 dB, RTRNLMS has -3.52 dB, DTNLMS has -6.77 dB, UBTNLMS has -7.96 dB and for sparse impulse response the existing DFNLMS has the cepstral distance of -3.40 dB, RTRNLMS has -4.41 dB, DTNLMS has -6.46 dB, and UBTNLMS has -8.97 dB. Sparse impulse response performs better for all the algorithms.

From the measures shown in Tables 2, 3 and 4, while analysing the performances in dispersive impulse response and sparse impulse response, UBTNLMS outperforms all

other algorithms for sparse impulse response. Taking all the performance measure into account, while comparing the overall algorithms in both dispersive and sparse impulse responses, Sparse impulse response performance in UBTNLMS has improved much better than the dispersive impulse response in all the noise types because it automatically determines the locations of the nonzero impulse response coefficients for their adaptation and eliminates the unnecessary adaptation for zero coefficients.

6. Conclusion

In this work, we show that the proposed algorithms RTRNLMS, DTNLMS and UBTNLMS perform better than the existing DFNLMS algorithm. The existing algorithm has large number of processing steps for enhancing the speech signal. To overcome this problem, proposed algorithms are designed to improve the performance with reduced computational steps. Out of the three proposed algorithms, the UBTNLMS algorithm shows good behaviour of superiority for both dispersive impulse response and sparse impulse response. In specific, the UBTNLMS shows better performance for sparse impulse response. The

Table 3 Performance of CD for the existing and proposed algorithms in terms of dispersive and sparse impulse responses

| Noise type | Input SNR (dB) | Cepstral distance (CD) in dB | | | | | | | |
|------------|----------------|------------------------------|---------------------|----------|--------|-------------------------|---------------------|----------|----------------|
| | | Dispersive impulse response | | | | Sparse impulse response | | | |
| | | Existing DFNLMS | Proposed algorithms | | | Existing DFNLMS | Proposed algorithms | | |
| | | RTR NLMS | DT NLMS | UBT NLMS | | RTR NLMS | DT NLMS | UBT NLMS | |
| Station | 0 | - 2.49 | - 3.52 | - 6.77 | - 7.96 | - 3.40 | - 4.41 | - 6.46 | - 8.97 |
| | 5 | - 3.52 | - 4.72 | - 7.27 | - 8.03 | - 4.43 | - 5.77 | - 7.21 | - 9.09 |
| | 10 | - 4.82 | - 5.97 | - 7.38 | - 8.99 | - 5.91 | - 6.89 | - 7.38 | - 10.31 |
| | 15 | - 4.82 | - 5.97 | - 7.38 | - 9.51 | - 7.38 | - 8.12 | - 8.83 | - 10.42 |
| Street | 0 | - 2.37 | - 3.30 | - 7.09 | - 7.62 | - 3.33 | - 4.28 | - 7.10 | - 8.66 |
| | 5 | - 3.34 | - 4.26 | - 6.81 | - 8.52 | - 4.31 | - 5.32 | - 6.89 | - 9.57 |
| | 10 | - 5.11 | - 5.94 | - 6.81 | - 9.73 | - 6.20 | - 6.81 | - 7.14 | - 10.54 |
| | 15 | - 6.37 | - 7.45 | - 7.28 | - 9.48 | - 6.90 | - 7.08 | - 8.14 | - 10.49 |
| Car | 0 | - 2.31 | - 3.28 | - 6.43 | - 7.54 | - 3.16 | - 4.26 | - 6.45 | - 8.56 |
| | 5 | - 3.80 | - 4.81 | - 7.05 | - 8.39 | - 4.76 | - 5.76 | - 7.18 | - 9.35 |
| | 10 | - 5.04 | - 6.56 | - 7.45 | - 8.99 | - 5.77 | - 7.28 | - 7.30 | - 10.08 |
| | 15 | - 6.18 | - 7.30 | - 7.49 | - 9.08 | - 7.20 | - 8.48 | - 9.79 | - 10.08 |
| Babble | 0 | - 2.93 | - 3.44 | - 6.17 | - 7.85 | - 3.94 | - 4.47 | - 7.17 | - 8.84 |
| | 5 | - 3.91 | - 4.70 | - 7.13 | - 8.87 | - 5.31 | - 5.89 | - 7.20 | - 9.86 |
| | 10 | - 4.98 | - 5.52 | - 7.17 | - 9.26 | - 6.03 | - 6.60 | - 8.33 | - 10.24 |
| | 15 | - 5.92 | - 7.09 | - 7.42 | - 9.27 | - 7.12 | - 7.19 | - 9.19 | - 10.28 |

The bold in the last column of Table is the indication and quantitative evidence of superior performance of UBT NLMS algorithm over RTR NLMS and DT NLMS algorithms in speech enhancement

Table 4 Performance of MSE for the existing and proposed algorithms in terms of dispersive and sparse impulse responses

| Noise type | Input SNR (dB) | Mean square error (MSE) in dB | | | | | | | |
|------------|----------------|-------------------------------|---------------------|----------|----------|-------------------------|---------------------|----------|-----------------|
| | | Dispersive impulse response | | | | Sparse impulse response | | | |
| | | Existing DFNLMS | Proposed algorithms | | | Existing DFNLMS | Proposed algorithms | | |
| | | RTR NLMS | DT NLMS | UBT NLMS | | RTR NLMS | DT NLMS | UBT NLMS | |
| Station | 0 | - 65.90 | - 73.17 | - 98.55 | - 98.34 | - 73.58 | - 80.94 | - 106.24 | - 116.99 |
| | 5 | - 73.32 | - 87.16 | - 99.47 | - 110.65 | - 81.33 | - 91.94 | - 106.97 | - 117.14 |
| | 10 | - 84.30 | - 91.91 | - 107.82 | - 117.82 | - 92.71 | - 101.17 | - 116.27 | - 126.38 |
| | 15 | - 94.66 | - 101.47 | - 110.64 | - 120.54 | - 102.69 | - 111.20 | - 119.80 | - 129.71 |
| Street | 0 | - 64.36 | - 71.78 | - 94.95 | - 107.05 | - 71.84 | - 79.65 | - 103.65 | - 114.73 |
| | 5 | - 72.44 | - 80.94 | - 104.04 | - 114.21 | - 82.15 | - 89.66 | - 112.41 | - 122.64 |
| | 10 | - 87.32 | - 92.49 | - 111.34 | - 121.34 | - 94.13 | - 101.01 | - 119.24 | - 129.37 |
| | 15 | - 91.99 | - 100.84 | - 111.55 | - 121.39 | - 103.97 | - 111.94 | - 120.09 | - 130.02 |
| Car | 0 | - 63.13 | - 72.33 | - 94.71 | - 104.30 | - 71.13 | - 80.00 | - 102.88 | - 113.97 |
| | 5 | - 75.38 | - 85.05 | - 102.00 | - 112.18 | - 83.69 | - 93.50 | - 110.10 | - 120.28 |
| | 10 | - 85.08 | - 98.27 | - 110.24 | - 120.22 | - 92.69 | - 105.52 | - 115.31 | - 125.73 |
| | 15 | - 93.48 | - 103.18 | - 108.66 | - 118.56 | - 103.64 | - 113.69 | - 116.56 | - 126.56 |
| Babble | 0 | - 69.96 | - 74.15 | - 96.38 | - 107.19 | - 77.11 | - 80.83 | - 105.23 | - 115.77 |
| | 5 | - 78.41 | - 86.67 | - 106.07 | - 116.19 | - 86.58 | - 94.34 | - 114.18 | - 124.91 |
| | 10 | - 86.90 | - 89.85 | - 108.57 | - 118.91 | - 95.30 | - 98.32 | - 116.87 | - 127.46 |
| | 15 | - 94.69 | - 106.50 | - 110.62 | - 120.57 | - 102.03 | - 110.09 | - 117.72 | - 127.64 |

The bold in the last column of Table is the indication and quantitative evidence of superior performance of UBT NLMS algorithm over RTR NLMS and DT NLMS algorithms in speech enhancement

algorithms are tested under the objective criteria like SNR, MSE, SegMSE and CD. For all the objective criteria, UBTNLMS performs better in both sparse and dispersive impulse response.

References

- [1] L Lin et al *Electron. Lett.* **39** 754 (2003)
- [2] D L Sun and G J Mysore *IEEE Acoust. Speech Signal Process. (ICASSP)* **141** (2013)
- [3] P S R Diniz Springer (2008)
- [4] M Djendi and P Scalart *Proc. IEEE. EUSIPCO* **1** 2080 (2012)
- [5] M Djendi, P Scalart and A Gilloire *Speech Commun. Elsevier* **55** 975 (2013)
- [6] M Djendi *Int. J. Adapt Control Signal Process* **29** 273 (2015)
- [7] M Djendi, R Bendoumia *Comput. Electr. Eng.* **39** 2531 (2013)
- [8] Y Ephraim and D Malah *IEEE Trans. Acoust. Speech Signal Process* **33** 443 (1985)
- [9] S Boll *IEEE Trans. Acoust. Speech Signal Process* **27** 113 (1979)
- [10] Y Ephraim and H L Van Trees *IEEE Trans. Speech Audio Process* **3** 251 (1995)
- [11] P Scalart *IEEE Acoust. Speech Signal Processing* **2** 629 (1996)
- [12] K Hermus and P Wambacq *EURASIP J. Adv. Signal Process* **1** (2007)
- [13] Y Ephraim and H L Van Trees *IEEE Trans. Speech Audio Process* **3** 251 (2006)
- [14] B De Moor *IEEE Trans. Signal Process* **41** 2826 (1993)
- [15] M Djendi, A Gilloire and P Scalart *Proc EUSIPCO* **1** 218 (2007)
- [16] Nguyen and Jutten *Signal Process* **45** 209 (1995)
- [17] M Djendi *Comput. Electr. Eng.* **52** 12 (2016)
- [18] M Djendi, A Gilloire and P Scalart *IEEE Int. Conf. ICASSP* **3** 744 (2006)
- [19] M Djendi and P Scalart *Proc. IEEE EUSIPCO* **2080** (2012)
- [20] P S R Diniz, G O Pinto and A Hjørungnes *Proc. ICASSP* **3833** (2008)
- [21] M Zoulikha and M Djendi *Appl. Acoust.* **112** 192 (2016)
- [22] Bendoumia R and Djendi M *Appl. Acoust.* **137** 69 (2018)
- [23] M Djendi and R Bendoumia *Appl. Soft Comput.* **42** 132 (2016)
- [24] M Djendi, R Henni and A Sayoud *International conference on engineering and MIS ICEMIS* (2016)
- [25] E Weinstein, M Feder, and A V Oppenheim *IEEE Trans. Speech Audio Process* **1** 405 (1993)
- [26] S Van Gerven and D Van Compernelle *Eur. Signal Process. Conf* **1** 1081 (1992).
- [27] S Van Gervan and D Van Compernelle *IEEE Trans. Signal Process* **43** 1602 (1995)
- [28] A Sayoud, M Djendi and *Appl. Acoust.* **135** 101 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.