

Fast Adaptive Filtering Algorithm for Acoustic Noise Cancellation

M. Arezki, A.Namane, A. Benallal, P. Meyrueis and D. Berkani

Abstract— In this paper, we propose an algorithm based on the adaptive filtering which we can use for the noise cancellation. This algorithm is stable numerically and very powerful when the input signal is stationary with additive noise on the output signal. It has a high convergence speed comparable with that of RLS algorithm and reduced computational complexity close to NLMS algorithm. The proposed algorithms allow a greater choice of compromise between these performances criteria and the computational complexity.

Index Terms— NLMS, Fast RLS, FTF-type algorithm, Estimation, Adaptive Filtering, ANC.

I. INTRODUCTION

ACOUSTIC noise cancellation (ANC) techniques are usually applied in applications where a reference signal that is correlated with the noise at the primary signal is easily obtained [1]. In these techniques, the reference signal is uncorrelated with the clean speech signal. These techniques make use of noise reference input and attempt to subtract the noise component from the noisy speech signal. The primary microphone picks up the noisy speech signal while a set of secondary microphone measure a signal consisting mainly of noise. The signal from the reference input is fed to an adaptive filter which estimates the noise on the primary input and simply subtracts it from the transmitted speech [2]. A number of filter structures and adaptation algorithms have been evaluated in the literature. There are two major classes of adaptive algorithms [3]. One is the Normalized Least Mean Square (NLMS) algorithm, which has a computational complexity of $O(L)$, L is the finite impulse response (FIR) filter length. The other class of adaptive algorithm is the Recursive Least Squares (RLS) algorithm has an impressive performance. The main drawback with the RLS algorithm is its complexity $O(L^2)$. A

large number of fast RLS (FRLS) algorithms have been developed over the years [4]-[6], but, unfortunately, it seems that the better a FRLS algorithm is in terms of computational efficiency, the more severe is its problems related to numerical stability. The FRLS algorithm shows a complexity of $O(L)$. Several numerical solutions of stabilization, with stationary signals, are proposed in the literature [7]. A new technique using reduced size predictors of order P , where $P \ll L$, was developed in [8], [9]. Fast RLS algorithms that use this technique are often called Fast Newton Transversal Filter (FNTF) Algorithms. Fast RLS algorithms that use this technique are often called Fast Newton Transversal Filter (FNTF) Algorithms. Several fast RLS algorithms including [10], [11] use the technique, and they are able to produce fast stable RLS algorithms that have a complexity of $(2L+4P) \approx 2L$. The computational complexity of these algorithms approach that of NLMS, and ought to be a very attractive choice for implementation.

In many applications of noise cancellation, the changes of the signal characteristics can be made quickly. This requires the use of adaptive algorithms, which converge rapidly. Our main objective is to evaluate the adaptive performances of the algorithms in real situations with criteria measured by the mean square error $MSE(n)$ and the gain G_{SNR} according to SNR_{in} . The proposed algorithms provide better results when making the noise less audible at exit of the system of noise cancellation.

II. ACOUSTIC NOISE CANCELLATION

Conventional ANC employs two input signals to reduce the noise at the output of the system as illustrated in Fig 1. The primary input is the noise-corrupted signal d_n . The reference input x_n , is measure of background noise alone which is in some way correlated with the noise in the primary. The output a priori error $\bar{\epsilon}_{L,n}$ of this system at time n is:

$$\bar{\epsilon}_{L,n} = d_n - \hat{y}_n \quad (1)$$

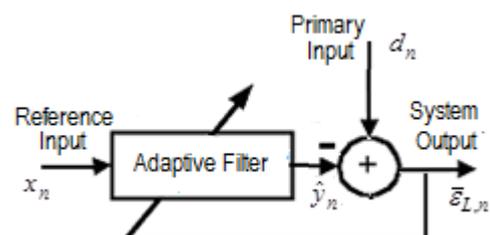


Fig 1: Principle of noise cancellation

Manuscript received March14, 2012; revised April16, 2012. This work was supported in part by the LATS1 Laboratory of the Department of Electronics, University of Blida Algeria and the LSP Laboratory, University of Strasbourg, France.

M. Arezki is with the Department of Electronics, University of Blida, B.P. 270 Route de Soumâa Blida 09000 Algeria. (phone: +213771204390; fax: +21325433850; e-mail: md_arezki@hotmail.com).

A.Namane is with the Department of Electronics, University of Blida, Algeria (e-mail: namane_a@yahoo.fr).

A. Benallal is with the Department of Electronics, University of Blida Algeria. (e-mail: a_benallal@hotmail.com).

P. Meyrueis is with the LSP Laboratory, University of Strasbourg, ENSPS, Bd. Sébastien Brant – B.P. 10413 ILLKIRCH 67412 France. (e-mail: meyrueis@spho.u-strasbg.fr).

D. Berkani, is with the Department of Electrical and Computer Engineering of ENP Algiers, 10 Avenue Pasteur, B.P. 182 El Harrach - 16000 Algiers Algeria. (e-mail: dberkani@hotmail.com).

where $\hat{y}_n = \mathbf{w}_{L,n-1}^T \mathbf{x}_{L,n}$ is the model filter output, $\mathbf{x}_{L,n} = [x_n, \dots, x_{n-L+1}]^T$ is a vector containing the last L samples of the input signal x_n , $\mathbf{w}_{L,n} = [w_{1,n}, \dots, w_{L,n}]^T$ is the coefficient vector of the adaptive filter and L is the filter length. The signal d_n is the contaminated signal (Fig 2) containing both the clean speech s_n and the noise y_n , assumed to be uncorrelated with each other. The primary input signal from the model is:

$$d_n = y_n + s_n \quad (2)$$

$$y_n = \mathbf{h}_n^T \mathbf{x}_{L,n} \quad (3)$$

where $\mathbf{h}_n = [h_{1,n}, \dots, h_{L,n}]^T$ represents the unknown system impulse response vector. The purpose of this system is to enable the system to control the filter until \hat{y}_n is as close to y_n as possible.

The error signal $\bar{\varepsilon}_{L,n}$ can be used to adapt the adaptive filter $\mathbf{w}_{L,n-1}$ using some algorithm for filter adaptation. Several different algorithms for filter adaptation have been proposed. The filter is updated at each instant by feedback of the estimation error proportional to the adaptation gain, denoted as $\mathbf{g}_{L,n}$, and according to

$$\mathbf{w}_{L,n} = \mathbf{w}_{L,n-1} + \mathbf{g}_{L,n} \bar{\varepsilon}_{L,n} \quad (4)$$

The different algorithms are distinguished by the adaptation gain calculation.

III. PROPOSED SYSTEM

A. Adaptive NLMS and FRLS Algorithms

The LMS Algorithms derived from the gradient [3], for which the optimization criterion corresponds to a minimization of the mean-square error. For the normalized LMS (NLMS) algorithm, the adaptation gain is given by:

$$\mathbf{g}_{L,n} = \frac{\mu}{L \pi_{x,n} + c_0} \mathbf{x}_{L,n} \quad (5)$$

where μ is referred to as the adaptation step and c_0 is a small positive constant used to avoid division by zero in absence of the input signal. The stability condition of this algorithm is $0 < \mu < 2$ and the fastest convergence is obtained for $\mu = 1$ [12]. The power of input signal is estimated

by $\pi_{x,n} = (1-\gamma) \pi_{x,n-1} + \gamma x_n^2$, where γ is a forgetting factor ($\gamma \approx 1/L$). The computational complexity of the NLMS algorithm is $2L$ multiplications per sample.

The RLS algorithm [3], for which minimizes a deterministic sum of squared errors. Fast versions of these algorithms are derived from the RLS by the introduction of forward and backward predictors. The FRLS algorithm shows a complexity of $O(L)$. The adaptation gain is given by:

$$\mathbf{g}_{L,n} = \underbrace{\mathbf{R}_{L,n}^{-1} \mathbf{x}_{L,n}}_{\text{RLS}} = \gamma_{L,n} \underbrace{\tilde{\mathbf{k}}_{L,n}}_{\text{FRLS}} \quad (6)$$

$$\mathbf{R}_{L,n} = \sum_{i=1}^n \lambda^{n-i} \mathbf{x}_{L,i} \mathbf{x}_{L,i}^T = \lambda \mathbf{R}_{L,n-1} + \mathbf{x}_{L,n} \mathbf{x}_{L,n}^T \quad (7)$$

where $\mathbf{R}_{L,n}$ is an estimate of the correlation matrix of the input signal vector and λ denotes the exponential forgetting factor ($0 < \lambda \leq 1$). The variables $\gamma_{L,n}$ and $\tilde{\mathbf{k}}_{L,n}$ respectively indicate the likelihood variable and normalized Kalman gain vector. Several numerical solutions of stabilization, with stationary signals, are proposed in the literature [7]. The computational complexity of the stabilized FRLS (NS-FRLS) algorithm is $8L$ per sample, and is stable, with the assumption of a white Gaussian input signal, under the following condition [7]:

$$\lambda > 1 - 1/2L \quad (8)$$

B. Proposed Adaptive Algorithms

The numerical stabilization of the FRLS algorithm limits the range of the forgetting factor (8) and consequently their convergence speed and tracking ability. And the resulting of that algorithm has an $8L$ complexity.

In [11], they propose more complexity reduction. The modified and simplified FTF-type (M-SMFTF) algorithm [11], derived from the SMFTF algorithm [10], where the adaptation gain is obtained only from the forward prediction variables and using a recursive method to compute the likelihood variable. The computational complexity is $6L$ and its algorithm is stable, with assumption of a white Gaussian input signal, under the following condition [11]:

$$\lambda > 1 - 1/L \quad (9)$$

By a method of extrapolation [9], the autocorrelation matrix of order L is built starting from an estimate of the autocorrelation matrix of order P ($P \ll L$). In this case, it is not anymore necessary to propagate prediction vectors of order L in the prediction part of the FRLS algorithm. If we denote P the order of the predictor and L the size of adaptive filter, the M-SMFTF algorithm can be easily used with reduced size prediction part. The computational complexity of the reduced size predictor M-SMFTF (RM-SMFTF) algorithm is $(2L + 4P) \approx 2L$ and it is stable, with the assumption of a white Gaussian input signal, under the following condition [11]:

$$\lambda > 1 - 1/P \quad (10)$$

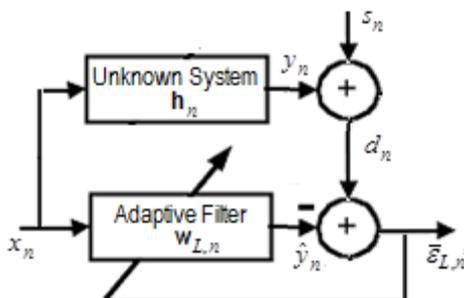


Fig 1: Principle of noise cancellation

IV. SIMULATIONS

The noise reference input pass through the adaptive filter and output \hat{y}_n is produced as close a replica as possible of y_n . The filter readjusts itself continuously to minimize the error between y_n and \hat{y}_n during this process. The system output is:

$$\bar{\varepsilon}_{L,n} = s_n + y_n - \hat{y}_n \quad (11)$$

We can get the following equation of expectations:

$$E\{(\bar{\varepsilon}_{L,n})^2\} = E\{(s_n + y_n - \hat{y}_n)^2\} \quad (12)$$

Assuming that x_n and s_n are not correlated and have zero means, we can write:

$$E\{(\bar{\varepsilon}_{L,n})^2\} = E\{(s_n)^2\} + E\{(y_n - \hat{y}_n)^2\} \quad (13)$$

After convergence of the adaptive filter, explicitly,

$$\mathbf{w}_{L,n} \rightarrow \mathbf{h}_n \Rightarrow \hat{y}_n \rightarrow y_n \Leftrightarrow \bar{\varepsilon}_{L,n} \rightarrow s_n \quad (14)$$

A. Tracking Ability and Convergence

In the absence of the speech signal ($s_n=0$). The expression (13) becomes:

$$E\{(\bar{\varepsilon}_{L,n})^2\} = E\{(y_n - \hat{y}_n)^2\} \quad (15)$$

The input signal x_n used in our simulation is a white Gaussian noise, with mean zero and variance equal to 0.32. The impulse response of the system represents a real impulse response measured in a car and truncated to 256 samples. We compare the convergence speed and tracking capacity of the (NLMS, NS-FRLS, M-SMFTF and RM-SMFTF) algorithms. The filter length is $L=256$, the NLMS ($\mu=1$) and NS-FRLS ($\lambda=1-1/3L$) algorithms are tuned to obtain fastest convergence. The forgetting factor λ are respectively chosen for M-SMFTF and RM-SMFTF algorithms to $\lambda=1-1/L$ and $\lambda=1-1/P$. The nonstationarity of the system to be modelled is simulated by

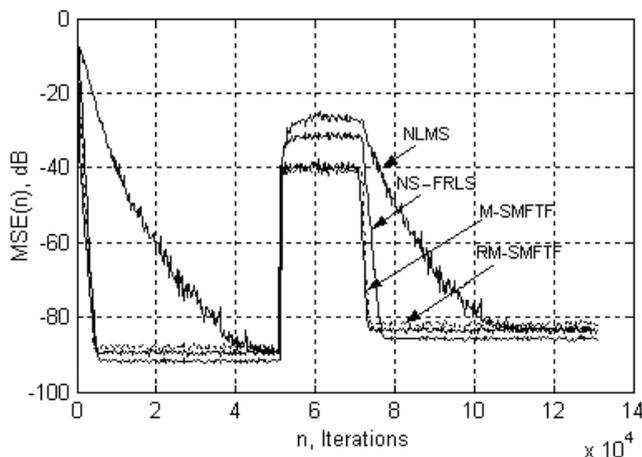


Fig 3: Comparative performance of the algorithms,

$L=256$. M-SMFTF: $\lambda=0.9961, \eta=0.985, c_a=0.5, E_0=1$;

RM-SMFTF: $P=32, \lambda=0.9688, \eta=0.9985, c_a=0.5, E_0=0.2$;

NS-FRLS: $\lambda=0.9987, E_0=1$; NLMS: $\mu=1$

introducing a linear gain variation on the primary input signal. In Fig 3, we give the evolution, in decibels, of the mean square error $MSE(n)$ (15). It shows that better performances in convergence speed are obtained for the proposed algorithm. It is observed that the proposed algorithm converges much faster and tracks better the variation of the system than both NS-FRLS and NLMS algorithms.

B. Noise Reduction

We define the input signal to noise ratio segmental [13], which is calculated on screens of a few milliseconds:

$$SNR_{In} = \frac{1}{M} \sum_{k=0}^{M-1} 10 \log_{10} \left(\frac{\sum_{i=0}^{N-1} s_{i+kN}^2}{\sum_{i=0}^{N-1} y_{i+kN}^2} \right) \quad (16)$$

where N is the size of the screen and M the number of screens. Same manner, we define the output signal to noise ratio segmental:

$$SNR_{Out} = \frac{1}{M} \sum_{k=0}^{M-1} 10 \log_{10} \left(\frac{\sum_{i=0}^{N-1} s_{i+kN}^2}{\sum_{i=0}^{N-1} (s_{i+kN} - \hat{s}_{i+kN})^2} \right) \quad (17)$$

One of the qualifying criteria of the noise reduction algorithms is the increase measurement in signal to noise ratio. This criterion, noted G_{SNR} and expressed in dB, is defined as the difference between SNR_{out} and SNR_{in} for each screen, averaged on the whole of the screens:

$$G_{SNR} = SNR_{out} - SNR_{in} \quad (18)$$

We generate noisy speech signals with different SNR_{in} starting from files from clean speech signal alone and noise alone. We synthesize the noisy speech signals by adding the noise to the clean speech signal so as to reach the desired levels of SNR_{in} . In our simulations, we took in the case of the NS-FRLS algorithm, $\lambda=1-1/10L$ to ensure numerical stability.

The noisy speech signal with $SNR_{in} = 30$ dB represents a case which we can say without noise (Fig 4). We observe (Fig 5a), that the output error of the system, which represents the estimated speech signal ($\bar{\varepsilon}_{L,n} \rightarrow \hat{s}_n$), is almost confused with the original speech signal except for the zones of silence (very weak power). We can say in these zones of silence, in absence of noise the NS-FRLS algorithm adapts less than the other algorithms. The curves (b) and (c) of Fig.5, which represent the temporal evolutions SNR_{out} and G_{SNR} with $SNR_{in} = 30$ dB, confirm well that the NS-FRLS algorithm adapts less than the other algorithms. In Fig 6, with $SNR_{in} = -5$ dB, we notice that the speech signal is relatively drowned in the noise. Fig 7 gives the temporal evolutions SNR_{out} and G_{SNR} . We can say that the NS-FRLS algorithm is slightly above the proposed algorithms.

Fig 8 represents the gain G_{SNR} as a function of SNR_{In} . The original speech signal is combined with noise for SNR_{In} between -10dB and 20dB by steps of 5dB. We observe that, the gain G_{SNR} decrease with SNR_{In} , the system has an overall tendency to not to remove the noise when SNR_{In} is strong and especially for the NS-FRLS algorithm. It is not necessarily a handicap; values beyond 15dB correspond relatively to slightly disturbed conditions, and attenuate the noise does not systematically improve comfort of listening. We notice that the NS-FRLS algorithm is placed slightly above the proposed algorithms.

V. CONCLUSION

From these performances criteria, measured by the mean square error $MSE(n)$ and the gain G_{SNR} , we have noticed that the proposed algorithms provide better results when making the noise less audible at exit of the system of noise cancellation. The simulations have shown that, the estimate of the noise have sensibly improved the adaptive performances in terms of noise reduction without distorting the speech signal. The NLMS algorithm is definitely less powerful than the other algorithms, and that the proposed algorithms allow a greater choice of compromise between these performances criteria and the computational complexity.

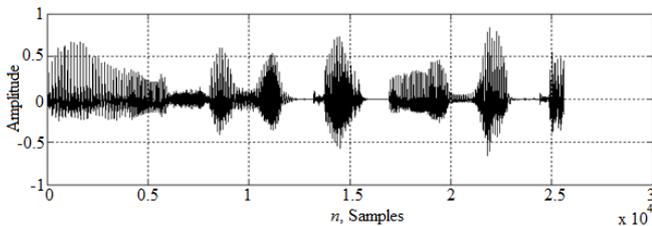


Fig 4 : Noisy speech signal with $SNR_{In} = 30dB$

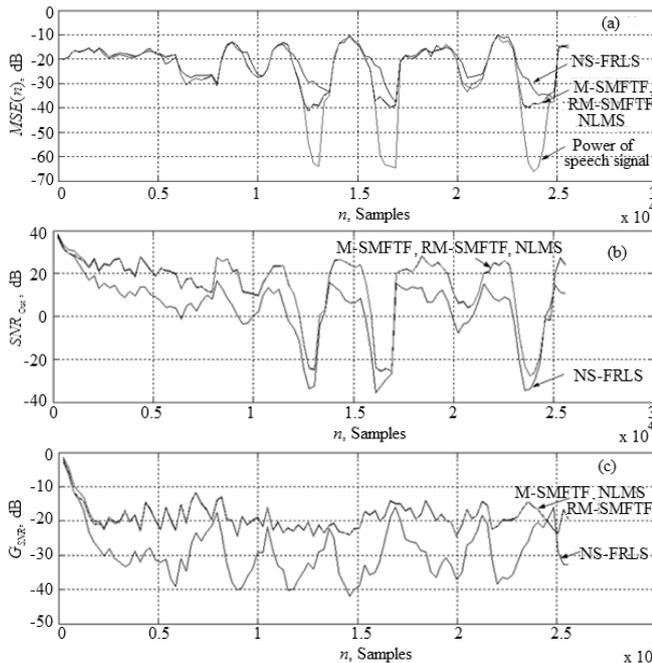


Fig 5 : Temporal evolutions: (a) $MSE(n)$; (b) SNR_{Out} ; (c) G_{SNR}

$L=256$. M-SMFTF: $\lambda = 0.9961, \eta = 0.985, c_a = 0.5, E_0 = 1$;
RM-SMFTF: $P=32, \lambda = 0.9688, \eta = 0.9985, c_a = 0.5, E_0 = 0.2$;
 λ μ

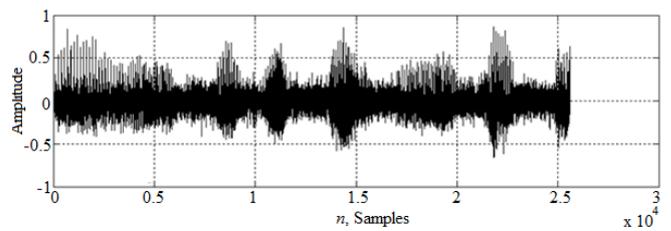


Fig 6 : Noisy speech signal with $SNR_{In} = -5dB$

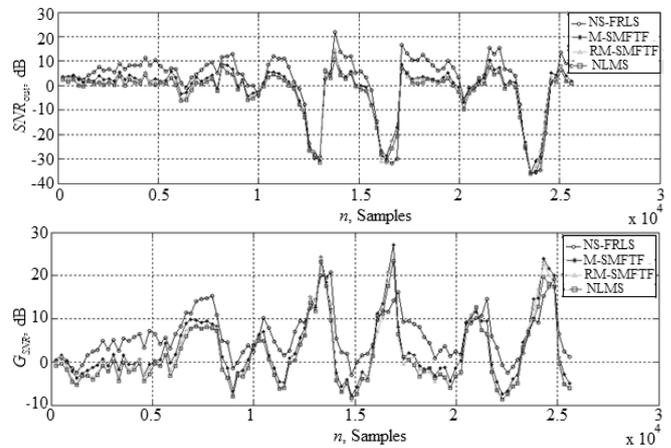


Fig 7 : Temporal evolutions of SNR_{Out} and G_{SNR} with $SNR_{In} = -5dB$.

$L=256$. M-SMFTF: $\lambda = 0.9961, \eta = 0.985, c_a = 0.5, E_0 = 1$;
RM-SMFTF: $P=32, \lambda = 0.9688, \eta = 0.9985, c_a = 0.5, E_0 = 0.2$;
NS-FRLS: $\lambda = 0.9996, E_0 = 0.5$; NLMS: $\mu = 1$.

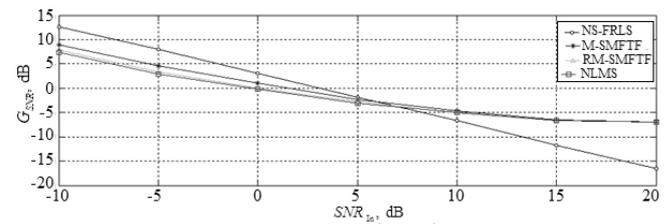


Fig 8 : $G_{SNR} = f(SNR_{In})$

$L=256$. M-SMFTF: $\lambda = 0.9961, \eta = 0.985, c_a = 0.5, E_0 = 1$;
RM-SMFTF: $P=32, \lambda = 0.9688, \eta = 0.9985, c_a = 0.5, E_0 = 0.2$;
NS-FRLS: $\lambda = 0.9996, E_0 = 0.5$; NLMS: $\mu = 1$.

REFERENCES

- [1] S. M. Kuo and D. R. Morgan, *Active Noise Control Systems*, New York: Wiley, 1996.
- [2] M. Arezki, P. Meyrueis, N. Javahiraly, "Specific signal processing method for sound detected by an intrinsic optical fiber sensor", Proc. SPIE 7675, 2010, 76750Q, Defense, Security, and Sensing, Orlando-USA, April 5-9, 2010.
- [3] S. Haykin, *Adaptive Filter Theory*, 4th ed., NJ, Prentice-Hall, 2002.
- [4] L. Ljung, M. Morf and D. Falconner, "Fast calculation of gain matrices for recursive estimation schemes," *Int. J. Control.*, vol. 27, 1978, pp.1-19.
- [5] G. Caryannis, D. Manolakis and N. Kalouptsidis, "A fast sequential algorithm for least squares filtering and prediction," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 31, 1983, pp.1394-1402.
- [6] J. Cioffi and T. Kailath, "Fast RLS Transversal Filters for adaptive filtering," *IEEE press. On ASSP*, vol.32, no.2, 1984, pp.304-337
- [7] M. Arezki, A. Benallal, P. Meyrueis, A. Guessoum and D. Berkani, "Error Propagation Analysis of Fast Recursive Least Squares Algorithms," *Proc. 9th IASTED International Conference on Signal and Image Processing*, Honolulu, Hawaii, USA, August 20-22, 2007, pp.97-101.
- [8] G.V. Moustakides and S. Theodoridis, "Fast Newton transversal filters - A new class of adaptive estimation algorithms," *IEEE Trans. Signal Process.*, vol.39, no.10, 1991, pp.2184-2193.

- [9] P. P. Mavridis and G. V. Moustakides, "Simplified Newton-Type Adaptive Estimation Algorithms," *IEEE Trans. Signal Process.*, vol.44, no.8, 1996.
- [10] A. Benallal and A. Benkrif, "A simplified FTF-type algorithm for adaptive filtering," *Signal processing*, vol.87, no.5, 2007, pp.904-917.
- [11] M. Arezki, A. Benallal, P. Meyrueis, and D. Berkani, "A New Algorithm with Low Complexity for Adaptive Filtering", *IAENG Journal, Engineering Letters*, vol.18, Issue 3, 2010, pp.205-211.
- [12] D.T.M. Slock, "On the convergence behaviour of the LMS and the NLMS algorithms," *IEEE Trans. Signal Processing*, vol.42, 1993, pp.2811-2825.
- [13] M. Bradstein and D. Ward, Eds., *Microphone arrays: Signal processing techniques and applications*. New York: Springer, 2001.