American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

http://asrjetsjournal.org/

Nonlinear Active Noise Control Using Adaptive Wavelet Filters

Mahdi Akraminia[a]*, Mohammad J. Mahjoob[b], Milad Tatari[c]

[a]*Department of Computer Engineering, Science and Research Branch Islamic Azad University, Tehran 1477893855, Iran*

[b]*School of Mechanical Engineering, College of Engineering, University of Tehran, Tehran 1417614418, Iran*

[c]*Department of Mechanical Engineering, University of Nevada, Reno 89557, USA*

[a]*Email: mahdi.akraminia@gmail.com*

[b]*Email: mmahjoob@ut.ac.ir*

[c]*Email: mtatari@unr.edu*

Abstract

This paper deals with nonlinear active noise control using adaptive wavelet filters. The ability of wavelets in signal reconstruction and function approximation make them appealing for black box system identification. Moreover, the intrinsic similarity between wavelet filters and noise/vibration signals implies that better approximation of these signals can be achieved by employing wavelet filters. Here, a new simple structure for using in active noise control system is proposed comprises a nonlinear static mapping cascaded with an IIR filter to take care of the dynamics of the system. With this strategy, one can avoid using multi-dimensional wavelet networks and thus eliminate curse of dimensionality. The performance of the proposed ANC system is examined for typical linear/nonlinear cases. The simulation results demonstrate superior performance of this method in terms of fast convergence rate and noise attenuation as well as computational complexity reduction while avoiding curse of dimensionality.

Keywords: Active noise control; Wavelet filters; Nonlinear system identification; Adaptive systems.

---

* Corresponding author.

1. Introduction

Acoustic noise problems are among the main concerns in the today's industrial world. The field of noise control can be broadly classified into two domains: passive noise control (PNC) and active noise control (ANC). PMC techniques aim at reducing the noise level by using absorbers and/or barriers. Although PNC method scan be efficient over a wide range of frequencies, their implementation at lower frequencies is costly while making the noise control system bulky. ANC systems have therefore been developed in order to overcome this limitation of PNC at low frequencies. Techniques used in ANC systems are based on the superposition of the primary noise source and the secondary source with their acoustic output shaving same amplitudes but opposite phases. The secondary source is powered by a controller, which utilizes a linear/nonlinear algorithm. A schematic diagram of a feedforward single cannel ANC system is shown in Figure 1. In ANC architecture, a reference microphone is used to feed the noise signal to the controller of the ANC system. The output of this controller is used to generate the anti-noise through a loudspeaker system. An error microphone is used to sense the residual noise which, in turn, is used to tune the ANC controller. The acoustic path from the noise source to the point of noise cancellation is called the *primary path*. The path from controller output through the loudspeaker, the acoustic path between the loudspeaker and the error microphone, plus the error microphone itself forms the *secondary path*.

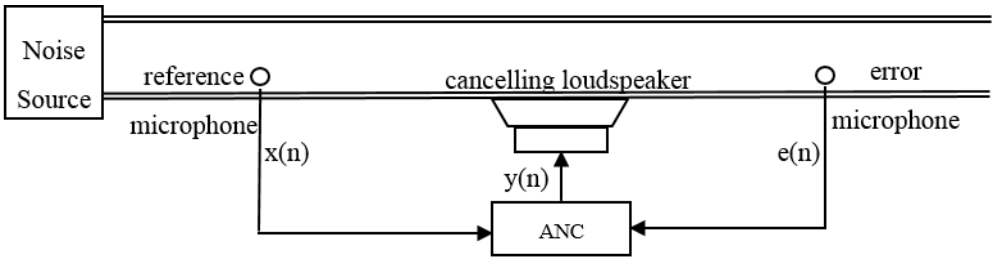


Figure 1: Feedforward ANC system in a duct

Linear ANC systems [1] have been successfully applied for active noise cancellation. However, their performance degrades where:

1- The noise coming from a dynamic system may be a nonlinear and deterministic noise process, such as a chaotic noise process rather than a stochastic, white, or tonal noise process [2], having a nonlinear term in the equation [3] and nonlinear noise inside an automotive cabin [4,5]
2- the secondary path transfer function between the speaker and the error microphone exhibits non-minimum phase response [2], or
3- the primary noise at the cancellation point exhibits nonlinear distortion due to nonlinear dynamics of the primary path (e.g. primary noise propagating in a duct with very high sound pressure) [2], or
4- saturation effects dominate, making the ANC system nonlinear. Saturation may occur in microphone and other electronic devices [6].

Nonlinear ANC systems overcome these limitation of linear ANC systems. The nonlinear approaches were first introduced by Synder and Tanaka [7] where they employed Multi-Layer Perceptron (MLP) network as general

function approximation in active noise control. Other nonlinear methods were then presented (reviewed by [8]) such as radial basis function networks [9], fuzzy neural networks [10], Volterra filters [11], functional link artificial, TT-transform [12], neural network [13-16], and convex combinations [17]. Modification were also made on using neural network to improve the performance of noise cancellation [18].

Families of frame wavelet functions are universal approximators. Inspired by capability of wavelets in signal reconstruction and function approximation, a novel approach is developed here based on wavelet filters including POLYWOG 1, RASP 1 and SLOG 1 wavelets [19-22].

Moreover, a new simple structure for using in active noise control system is proposed in which comprises a nonlinear static mapping cascaded with an IIR filter to take care of the dynamics of the system. In this structure, only input at the current step is required to generate the output of the next sample; therefore, there is no need to use tab delayed line of input-output of the physical system. With this strategy, one can avoid using multi-dimensional wavelet networks and thus eliminate curse of dimensionality.

2. Adaptive wavelet network controller for active noise control

2.1. Wavelet transform and wavelet families

A wavelet is a function $\psi \in L^2(R)$ with a zero average:

$$\int_{-\infty}^{\infty} \psi(t)\,dt = 0 \qquad (1)$$

It is normalized $\|\psi\| = 1$, and centered in the neighborhood of $t = 0$ [23]. A family of wavelets is obtained by scaling ψ by u and translating it by m:

$$\psi_{m,u} = \frac{1}{\sqrt{u}}\psi\left(\frac{t-m}{u}\right) \qquad (2)$$

The wavelet transform of $f \in L^2(R)$ at the time m and scale u is

$$Wf(m,u) = \left\langle f, \psi_{m,d} \right\rangle = \int_{-\infty}^{\infty} f(t)\frac{1}{\sqrt{d}}\psi^*\left(\frac{t-m}{u}\right)dt \qquad (3)$$

A real wavelet transform is complete and invertible as long as the wavelet satisfies a weak admissibility condition, specified by the following [23]:

Let $\psi \in L^2(R)$ be a real function such that

$$C_{\psi} = \int_{0}^{\infty} \frac{\left|\hat{\psi}(\omega)\right|}{\omega} d\omega < +\infty \qquad (4)$$

which is called wavelet admissibility condition. In equation (4), $\hat{\psi}(\omega)$ is the Fourier transform of $\psi(t)$ and ω represents frequency in the Fourier domain.

To guarantee that the wavelet transform is invertible, we must ensure that $\hat{\psi}(0) = 0$ and $\hat{\psi}(\omega)$ is continuously differentiable. Admissibility condition ensures any function $f \in L^2(R)$ can be represented by wavelets [23].

The family of functions generated by ψ can be defined as:

$$\Omega_c = \left\{ \frac{1}{\sqrt{u_j}} \psi\left(\frac{t - m_j}{u_j} \right), m_j \in R \,\&\, u_j \in d_j \right\} \qquad (5)$$

Two categories of wavelet functions, namely, orthogonal wavelets and wavelets frames, were developed. Orthogonal wavelet decomposition is usually associated to the theory of multidimensional analysis [24]. The fact that orthogonal wavelets cannot be expressed in closed form is a serious drawback for their application to function approximation. In contrast, wavelet frames called the mother wavelet, which must satisfy conditions that are less stringent than orthogonality conditions. A family Ω_c is said to be frame of $L^2(R)$ if there exist two constant $a > 0$ and $b > 0$ such that for any square integrable function f the following inequalities hold:

$$a\|f\|^2 \le \sum_{\psi_j \in \Omega_c} \left|\langle \psi_j, f \rangle\right|^2 \le b\|f\|^2 \qquad (6)$$

where $\|f\|$ denotes the norm of function f and $\langle f, g \rangle$ inner product of functions f and g. Families of wavelet frames of $L^2(R)$ are universal approximators [24].

2.1.1 POLYWOG wavelets

POLYnominal WinOwed with Gaussian (POLYWOG) wavelets are derivative of Gaussian function [25]. It is very simple to prove POLYWOG wavelets have the property $\hat{\psi}(0) = 0$ and $\hat{\psi}(\omega)$ is continuously differentiable. Therefore, they can be considered as wavelet frames. POLYWOG 1 wavelet filter which is used here is as follows:

$$\psi_{Polywog1} = k_1 z e^{-z^2/2} \qquad k_1 = 2/\sqrt{\pi} \qquad (7)$$

where k_1 is determined such that $\left\|\psi_{Polywog1}\right\|_2 = 1$. Figure 2 shows the POLYWOG 1.

2.1.2 RASP wavelets

Rational functions with Second-order Poles (RASP) arise from the residue theorem of complex variables [26]. These wavelets are real, odd functions with zero mean. The distinction among these mother wavelets is their rational form of strictly proper functions and having simple/double poles of second order.

According to the admissibility condition, it must be shown that the families of RASP mother wavelets have zero mean value. This can be done easily using the residue theorem for evaluation of integrals. RASP 1 wavelet which is used here is as follows:

$$\psi_{Rasp1} = \frac{k_2 z}{\left(z^2+1\right)^2} \qquad k_2 = 3.0788 \qquad (8)$$

where k_2 is determined such that $\left\|\psi_{Rasp1}\right\|_2 = 1$. Figure 2 demonstrates the RASP 1.

2.1.3 SLOG wavelets

This family of frame mothers are the Superposed LOGistic functions (SLOG). This taxon of mothers results from finite term sums of weighted and delayed logistic functions. A logistic function is a type of sigmoidal function having the form $1/1+e^{-t}$, a sigmoidal function is a strictly increasing, smooth, asymptotic function.

This family of wavelets arises from a specific domain, namely, the theory of neural networks. Here, a mother wavelet is constructed using weighted, delayed versions of the neural net activation function which is typically sigmoidal and logistic [26]. SLOG 1 wavelet which is used here is:

$$\psi_{Slog1} = k_3\left(\frac{1}{1+e^{-z+1}} - \frac{1}{1+e^{-z+3}} - \frac{1}{1+e^{-z+3}} + \frac{1}{1+e^{-z-1}}\right) \qquad k_3 = 1.0572 \qquad (9)$$

where k_3 is determined such that $\left\|\psi_{Slog1}\right\|_2 = 1$. Figure 2 depicts the SLOG 1.

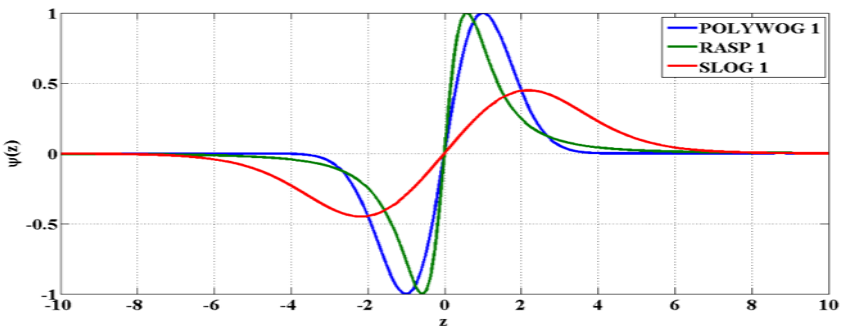


Figure 2: Wavelet filters used in this study

2.2 Adaptive wavelet network

In this subsection, the structure of Adaptive Wavelet Network (AWN) for active noise control is introduced (Figure 3).

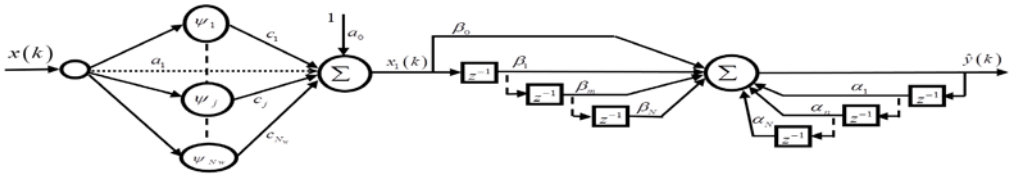


Figure 3: Architecture of proposed Adaptive Wavelet Network (AWN) based on POLYWOG wavelet frames

In Figure 3, $x(k)$ is the input signal, $x_1(k)$ is the output of nonlinear static mapping part and can be obtained as follows:

$$x_1(k) = \sum_{j=1}^{N_w} c_j \psi_j(z_j(k)) + a_1 x(k) + a_0 \qquad (10)$$

where $\psi_j(.)$ is the wavelet functions listed in table 1, and $z_j(k) = \dfrac{x(k) - m_j}{u_j}$ in which m_j is the translation parameter, u_j is the dilation parameter, a_0 refers to the bias coefficient, a_1 represents the direct connection weight and N_w is the total number of wavelet functions employed in the network. The output of our AWN is:

$$\hat{y}(k) = \sum_{j=0}^{N} \beta_m x_1(k-m) + \sum_{j=1}^{N} \alpha_n \hat{y}(k-n) \qquad (11)$$

where N is the order of system, β_m are the feedforward coefficients and α_n are the feedback coefficient of the recursive part. The AWN is designed for the modeling a more general nonlinear ANC system. It is comprised of two cascade section: a nonlinear static mapping based on wavelet frames and a recursive filter for modeling the dynamic part.

2.3 Structure of wavelet based ANC

The structure of the wavelet based active noise control is demonstrated schematically in Figure 4.

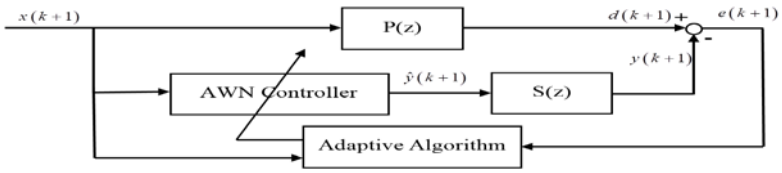


Figure 4: Structure of the wavelet based active noise control

P(z) represents the transfer function of primary path while S(z) indicates the transfer function of secondary path. The output of AWN controller $\hat{y}(k+1)$ drives the secondary path and secondary path generates the acoustic signal $y(k+1)$ given by:

$$y(k+1)=\sum_{j=0}^{M-1} s_j \hat{y}(k+1-j) \qquad (12)$$

where s_j is the coefficient of secondary path (known already from previous identification stage[1]). The aim of our adaptive algorithm is to update the AWN controller to minimize the residual noise $e(k+1)$ measured by the error microphone. The structure of adaptive algorithm is presented in the following subsection.

2.4 Online training of POLYWOG AWN

In this subsection, the training relations of the AWN controller are derived. The gradient descent algorithm is employed; however the secondary path transfer function between AWN controller and error microphone should be considered in the learning procedure. Modifications are therefore made in the gradient algorithm to be applied in AWN for active noise control.

To accomplish this we define a cost function as follows

$$J(k+1)=\frac{1}{2}e^2(k+1)=\frac{1}{2}\left(d(k+1)-y(k+1)\right)^2 \qquad (13)$$

which is minimized during the learning process by tuning the AWN controller parameters. Then note that the vector of parameters consists of

$$\underline{\theta}=\left[a_0,a_1,c_1...c_j...c_{N_w},m_1...m_j...m_{N_w},u_1...u_j...u_{N_w},\beta_0...\beta_m...\beta_N,\alpha_1...\alpha_n...\alpha_N\right]^T \qquad (14)$$

To minimize the cost function, the following updating relation is followed for every component of the vector $\underline{\theta}$ introduced in (14):

$$\theta_j(k+1)=\theta_j(k)+\Delta\theta_j(k+1)=\theta_j(k)-\mu\nabla J_{\theta_j}(k+1) \qquad (15)$$

where

$$\nabla J_{\theta_j}(k+1)=\frac{\partial J(k+1)}{\partial\theta_j}=-e(k+1)\cdot\frac{\partial y(k+1)}{\partial\theta_j} \qquad (16)$$

Now, our objective is to find the term $\frac{\partial y(k+1)}{\partial\theta_j}$ for every component of vector $\underline{\theta}$. Hence differentiate (12)

w.r.t. every component θ_j beginning from a_0 and a_1 up to α_n :

$$\frac{\partial y(k+1)}{\partial a_0} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial a_0} \qquad (17)$$

Using equation (11), we can write:

$$\begin{aligned}\frac{\partial \hat{y}(k+1-i)}{\partial a_0} &= \sum_{m=0}^{N} \beta_m \frac{\partial x_1(k+1-i-m)}{\partial a_0} + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial a_0} \\ &= \sum_{m=0}^{N} \beta_m + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial a_0} \end{aligned} \qquad (18)$$

Similarly differentiate (12) w.r.t. a_1 :

$$\frac{\partial y(k+1)}{\partial a_1} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial a_1} \qquad (19)$$

reuse (11):

$$\begin{aligned}\frac{\partial \hat{y}(k+1-i)}{\partial a_1} &= \sum_{m=0}^{N} \beta_m \frac{\partial x_1(k+1-i-m)}{\partial a_1} + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial a_1} \\ &= \sum_{m=0}^{N} \beta_m x\left(k+1-i-m\right) + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial a_1} \end{aligned} \qquad (20)$$

For the coefficients of wavelet nodes c_j , we have:

$$\frac{\partial y(k+1)}{\partial c_j} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial c_j} \qquad (21)$$

$$\begin{aligned}\frac{\partial \hat{y}(k+1-i)}{\partial c_j} &= \sum_{m=0}^{N} \beta_m \frac{\partial x_1(k+1-i-m)}{\partial c_j} + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial c_j} \\ &= \sum_{m=0}^{N} \beta_m \psi_j\left(k+1-i-m\right) + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial c_j} \end{aligned} \qquad (22)$$

For translation and dilation wavelet parameters m_j and u_j :

$$\frac{\partial y(k+1)}{\partial m_j} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial m_j} \qquad (23)$$

$$\frac{\partial \hat{y}(k+1-i)}{\partial m_j} = \sum_{m=0}^{N} \beta_m \frac{\partial x_1(k+1-i-m)}{\partial m_j} + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial m_j}$$

(24)

$$\frac{\partial y(k+1)}{\partial u_j} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial u_j} \qquad (25)$$

$$\frac{\partial \hat{y}(k+!-i)}{\partial u_j} = \sum_{m=0}^{N} \beta_m \frac{\partial x_1(k+1-i-m)}{\partial u_j} + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial u_j}$$
$$= \sum_{m=0}^{N} \beta_m \frac{-c_j \times z_j(k+1-i-m)}{u_j} \psi_j'\left(k+1-i-m\right) + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial u_j} \qquad (26)$$

For feedforward and feedback weights of recursive filter β_m and α_n:

$$\frac{\partial y(k+1)}{\partial \beta_m} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial \beta_m} \qquad (27)$$

$$\frac{\partial \hat{y}(k+1)}{\partial \beta_m} = x_1(k+1-i-m) + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial \beta_m} \qquad (28)$$

$$\frac{\partial y(k+1)}{\partial \alpha_n} = \sum_{i=0}^{m} s_i \frac{\partial \hat{y}(k+1-i)}{\partial \alpha_n} \qquad (29)$$

$$\frac{\partial \hat{y}(k+1-i)}{\partial \alpha_n} = \hat{y}(k+1-i-n) + \sum_{n=1}^{N} \alpha_n \frac{\partial \hat{y}(k+1-i-n)}{\partial \alpha_n} \qquad (30)$$

3. Neural network base nonlinear ANC algorithm

In order to compare the proposed algorithm with the neural network (NN) based nonlinear ANC, the Filtered-x Back Propagation Neural Network (FxBPNN) algorithm is briefly presented. In fact, FxBPNN is simply extension of the linear FxLMS algorithm in which the output layer has a single node and a linear nodal output function. It is also a form of gradient descent algorithm in which the error signal is used to adjust the weights in both the output and hidden layers.

The block diagram of the single-channel broadband feedforward ANC system using Neural Network (NN) algorithm is illustrated in Figure 5 and Figure 6. In Figure 5, $P(z)$ shows the primary path and $S(z)$ demonstrates the secondary path.

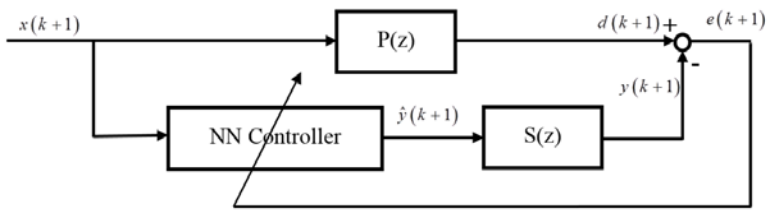


Figure 5: The block diagram of active noise control using the NN controller

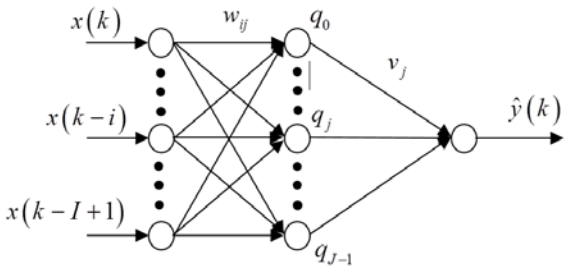


Figure 6: Structure of the NN controller

The aim of the NN controller is to minimize the cost function introduced in (13). To meet this end, FxBPNN has been developed in two parts, (A) forward and (B) backward propagation as follows [27]:

(A) Forward propagation:

Step (1): Input the reference signal $X\left(k+1\right)$ and the error signal $e\left(k+1\right)$ at time $k+1$, where $X\left(k+1\right)=\left[x\left(k+1\right)x\left(k\right)...x\left(k-I\right)\right]^T$ is the reference signal vector.

Step (2): Calculate the output of the hidden layer $q_j\left(k+1\right)$.

$$q_j\left(k+1\right)=f_j\left(net_j\left(k+1\right)\right)=\tanh_j\left(net_j\left(k+1\right)\right) \qquad (31)$$

where, $net_j\left(k+1\right)=\sum_{i=0}^{I-1}w_{ij}x\left(k+1-i\right)$.

$$\underline{\underline{W}}=\begin{bmatrix} w_{00} & w_{01} & ... & w_{0J-1} \\ w_{10} & w_{11} & ... & w_{1J-1} \\ ... & ... & w_{ij} & ... \\ w_{I-10} & w_{I-11} & ... & w_{I-1J-1} \end{bmatrix}$$

is the matrix of weights in the hidden layer and I is the number of inputs.

$q_j\left(k+1\right)$ is the output of the *j*th hidden layer neuron. $f\left(.\right)$ is the smoothing activation function which considered here as tangent hyperbolic function and J is the total number of the hidden layer neurons.

Step (3): Calculate the output of the output layer $\hat{y}\left(k+1\right)$ at time $k+1$:

$$\hat{y}(k+1)=\sum_{j=0}^{J-1} v_j(k+1)\,q_j(k+1) \qquad (32)$$

where $v_j(k+1)$, is the *j*th weight of output layer and J is the total number of weights in the output layer.

Step (4): Calculate the output of the system $y(k+1)$:

$$y(k+1)=\sum_{j=0}^{M-1} s_j\,\hat{y}_j(k+1-j) \qquad (33)$$

where s_j is the *j*th coefficient of secondary path model and M is the length of the secondary path.

(B) Back propagation:

In this stage, the weights of the hidden/output layers are updated in order to minimize the cost function introduced in (13). To meet this end, the gradient descent algorithm (introduced in (15)) is employed to adjust the weights of the control system. The adaptation algorithm of the weights in the output layer can be written as [27]:

$$v_j(k+1)=v_j(k)+\mu e(k)\sum_{i=0}^{M-1} s_i q_j(k-i),\quad j=0,1,...,J-1. \qquad (34)$$

while the adaptation algorithm of the weights in the hidden layer is [27]:

$$w_{ij}(k+1)=w_{ij}(k)+\mu e(k)\sum_{t=0}^{M-1} s_t v_j(k-t)\,f_j'\left(net_j(k-t)\right)x(k-i-t),\quad i=0,1,...,I-1,\ j=0,1,...,J-1. \qquad (35)$$

4. Simulation results and discussion

4.1. Simulation Results

In this section, results are presented to compare the performances of nonlinear ANC algorithm based on different wavelet filters presented in this study as well as Neural Network based ANC system. The number of wavelet neurons and order of IIR filter are also selected as $N_w=N=6$.The number of adaptive parameters, therefore, used in this structure is 33. The sampling frequency used in the simulations is 1000 Hz. Since a logistic chaotic noise is a second-order white noise and predictable nonlinear process [2], this signal along with an additional Gaussian white noise is used to generate the primary disturbance signal. The logistic chaotic noise is made using [2]

$$x(n+1)=\lambda x(n)(1-x(n)) \qquad (36)$$

where $\lambda=4$ and $x(0)=0.9$. The variance of Gaussian white noise is set to 0.02. The learning rate of our proposed method is set to 0.005.

Regarding the structure illustrated by the Figure 4 and Figure 5, two cases for $P(z)$ and $S(z)$ are considered. These cases have been selected based on the condition of an ANC system which may exhibit linear and/or nonlinear behavior.

Case 1. A linear system is considered. Therefore, the primary path and secondary path are selected as [23]:

$$d(k+1)=0.8x(k-6)+0.6x(k-7)-0.2x(k-8)-0.5x(x-9)-0.1x(k-10)+0.4x(k-11)-0.05x(k-12)$$
$$y(k+1)=0.9\hat{y}(k-2)+0.6\hat{y}(k-3)+0.1\hat{y}(k-4)-0.4\hat{y}(x-5)-0.1\hat{y}(k-6)+0.2\hat{y}(k-7)+0.1\hat{y}(k-8)+0.01\hat{y}(k-9)+0.001\hat{y}(k-10)$$

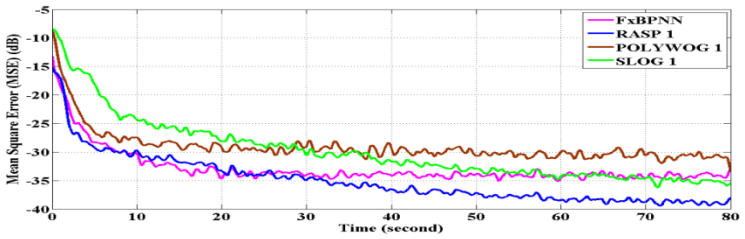
Although the primary path is chosen linear in this case, the overall system is nonlinear since logistic chaotic noise is a nonlinear process [6].

In Figure 7(a) and 7(b), the dash-dotted dark line is used when ANC is off, solid-magenta line depicts FxBPNN, solid-blue line shows the results of RASP 1, solid-brown line demonstrates the result of POLYWOG 1, solid-green represents the output of SLOG 1.

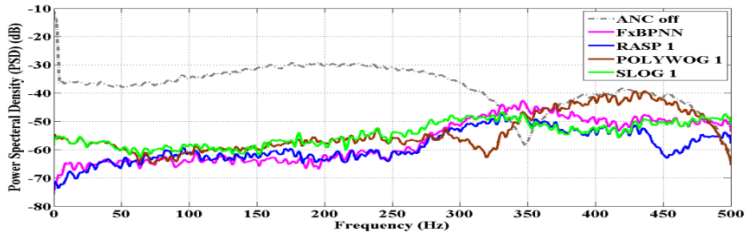
Figure 7(a) shows the convergence rate of proposed methods with different wavelet filters based on the Mean Square of cancelling Error (MSE) in dB versus time. In this case the convergence rate of RASP 1 is relatively improved; while POLYWOG 1 shows less attenuation for the same noise signal.

In order to show the performance of algorithms in the frequency domain, Figure 7(b) is presented which shows the Power Spectral Density (PSD) of square error of wavelet filters. In computing PSD, the last 5 seconds of the signal length was employed which represents the steady state response of the algorithms. As it can be seen in this Figure, improved attenuation of the logistic chaotic noise was obtained by RASP 1.

Case 2. The reference and error sensors of ANC system saturate if the noise level exceeds the dynamic range of the sensors. In this case, clipping signals of error and reference microphone are used to approximate the saturation. The theoretical analysis shows that the clipping of a sinusoidal signal produces extra odd harmonics, thus affecting the convergence speed [6]. The saturated signal is generated by defining a fixed threshold value (0.6 in this study) and clipping the signal if its absolute magnitude exceeds the threshold value. The primary path $P(z)$ and secondary path $S(z)$ are considered the same as case 1. Figure 8(a) and 8(b) show experimental results for saturation effect in the ANC system. The same line notation as in the previous case is used here. Figure 8(a) shows the MSE of cancelling error in dB versus time. SLOG 1 has a rather acceptable convergence rate while the RASP 1 maintains its superior convergence.Figure 8(b), shows PSD of cancelling error in dB in the frequency domain which implies improved attenuation by using RASP 1.

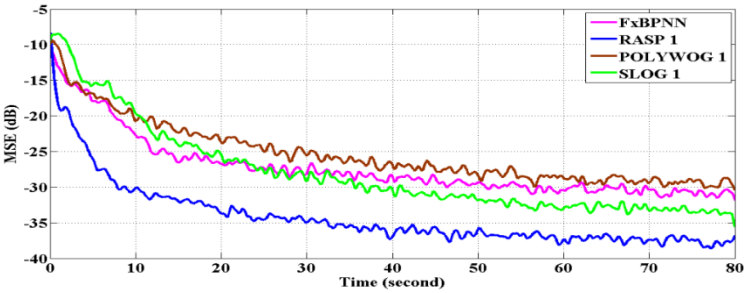


(a)

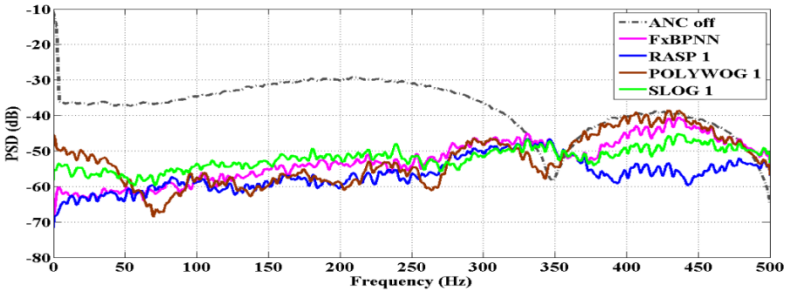


(b)

Figure 7: The dash-dotted dark curve is used when ANC is off, solid magenta depicts FxBPNN, solid-blue curve shows RASP 1 performance, solid-brown curve demonstrates POLYWOG 1 and solid-green represents SLOG 1. (a) Mean square of cancelling error (dB) versus time (second) where P(z) and S(z) are considered linear (b) Power Spectral Density (PSD) (dB) versus frequency (Hz) where P(z) and S(z) are considered linear



(a)



(b)

Figure 8: The dash-dotted dark curve is used when ANC is off, solid magenta depicts FxBPNN, solid-blue curve shows RASP 1 performance, solid-brown curve demonstrates POLYWOG 1 and solid-green represents SLOG 1. (a) Mean square of cancelling error (dB) versus time (second) where nonlinear P(z) is considered (b) Power Spectral Density (PSD) (dB) versus frequency (Hz) where nonlinear P(z) is considered

4.2. Computational load

In this subsection, the computational load of the algorithms are presented. For computational complexity per iteration of the AWN POLYWOG, let us consider N_w wavelet frames employed in the proposed method and N memory block in the IIR section. M is assumed as the length of secondary path transfer function. The computational complexity per iteration of our proposed algorithm is computed as follows:

i) For calculating the output of the AWN POLYWOG controller $\hat{y}(n)$:

Number of required multiplications = $2N+2N_w+2$, Number of required additions = $2N+2N_w+3$

Number of required wavelet frames and their derivatives = N_w

ii) For updating weights :

Number of required multiplications = $4N+6N_w+M(17N+10)+6$, Number of required additions = $2N+3N_w+M(12N+7)+3$

Number of required wavelet frames and their derivatives = N_w

iii) Total computational load

Total number of required multiplications = $6N+8N_w+M(17N+10)+8$

Total number of required additions = $4N+5N_w+M(12N+7)+6$, Total number of required wavelet frames and their derivatives = $2N_w$

Now, assuming N_1 inputs to the input layer of neural network, N_2 hidden nodes and similar to the previous case M is considered as the length of secondary path transfer function, the computational load per iteration of the neural network algorithm is calculated as follows:

i) For calculating the output of the neural network controller $\hat{y}(n)$:

Number of required multiplications = $N_1N_2+2N_2$, Number of required additions = $N_1N_2+2N_2$

Number of required wavelet frames and their derivatives = N_2

ii) For updating weights :

Number of required multiplications = $3MN_2\left(1+3N_1\right)$

Number of required additions = $MN_2\left(1+N_1\right)+N_1+2N_2$

Number of required wavelet frames and their derivatives = N_1N_2

iii) Total computational load

Total number of required multiplications = $3MN_2\left(1+3N_1\right)+N_1N_2+2N_2$

Total number of required additions = $MN_2\left(1+N_1\right)+N_1+4N_2+N_1N_2$

Total number of required activation function and its derivative = $N_2\left(1+N_1\right)$

The summary of computational complexity per iteration of algorithms for the cases are presented in the Table 1.

Table 1: Computational load per iteration of algorithms for cases

	AWN POLYWOG			**Neural Network**		
	Multiplication	addition	nonlinear function	Multiplication	addition	nonlinear function
Case 1	$N=N_w=6,\ M=9$			$N_1=N_2=10,\ M=9$		
	1100	771	12	8490	1140	110
	493	342	12	2910	480	110
Case 2	$N=N_w=6,\ M=9$			$N_1=N_2=10,\ M=9$		
	1100	771	12	8490	480	110
	1100	771	12	8490	480	110

Total number of adaptive parameters in our proposed method and the neural network approach are 33 and 110 respectively. As a result, computational complexity in the table 2 shows that the minimum computational load belongs to our proposed algorithm. Moreover, one can compare the required complexity for the AWN POLYWOG and neural network method assuming that the computational complexity of nonlinear functions of these two are equal. The number of required multiplication of AWN POLYWOG is less than 17% of neural network approach. The required number of additions for our proposed method is also less than 72% of neural network. Furthermore, the number of required nonlinear function calculation in the AWN POLYWOG is 10%

of the corresponding required calculation of nonlinear function in the neural network approach.

4.3 Discussion

The results obtained are discussed here. For every case of simulation results, two figures are presented. The first one shows the error signal in time domain while the second one demonstrates the error signal in the frequency domain. The aim of presenting the error signal in dB in time domain is to show the overall noise reduction of each algorithm and their convergence rate. The main purpose of demonstrating the error signal in the frequency domain is to show the performance of each algorithm in different frequency bands. There are some points which should be noted here. Similarities between wavelet frames and noise/ vibration signals imply that these frames can be good candidates for signal approximation of these variables. This is the main idea which motivated this study. This similarity can result in reducing number of required frames for signal approximation which this in turn decreases computational load as shown in this study. Among the adaptive parameters in the proposed network, the dilation parameters d_j are the most sensitive coefficients. Drastically changing of this parameter can make the network unstable. Therefore, the step size of these parameters are suggested to be 10% of the step size of translation parameters m_j . As a recommendation for future work, this method for active noise cancellation can be implemented when there are multiple noise sources in the system.

5. Conclusion

In this study, a nonlinear adaptive wavelet filters with moderately simple structure was introduced. In this novel structure a nonlinear static mapping cascaded with an IIR filter was utilized in order to model system dynamics. A highlighted advantage of the proposed structure was that only the input at the current step was required to generate the output of next sample. This eliminated the need for tab delayed line of input-output of the physical system. Furthermore, utilizing this strategy, one could avoid using multi-dimensional wavelet networks and hence the curse of dimensionality. In addition, online dynamic backpropagation learning algorithms based on gradient descent method were employed to update the parameters of Adaptive Wavelet Networks (AWN). The performance of the new algorithm was assessed using 3 wavelet filters while compared to the FxBPNN algorithm by considering 2 typical cases. Experiments revealed RASP 1 exhibited good attenuation and fast convergence rate among other wavelet filters while POLYWOG 1 presented poor outcomes. The simulation results also demonstrated superior performance of this method in terms of fast convergence rate and noise attenuation as well as computational complexity reduction while avoiding curse of dimensionality.

References

[1] S. M. Kuo, D. R. Morgan, “Active noise control: a tutorial review,” Proceedings of the IEEE, vol. 87, pp. 943-973, 1999.

[2] L. Tan, J. Jiang, “Adaptive Volterra filters for active control of nonlinear noise processes,” Signal Processing, IEEE Transactions on, vol. 49, pp. 1667-1676, 2001.

[3] N. Nasrollahzadeh, M. Fard, M. Tatari, "Parametric resonance: Application on low noise mechanical and electromechanical amplifiers," in Nonlinear Approaches in Engineering Applications 2, R. Jazar, L. Dai, Ed. New York: Springer, 2014, pp. 277-308.

[4] M. Tatari, M. Fard, N. Nasrollahzadeh, M. Mahjoob. "Nonlinear Vehicle Seat BSR Characterization Using CAE Methodology" in Nonlinear Approaches in Engineering Applications 2, R. Jazar, L. Dai, Ed. New York: Springer, 2014, pp. 231-256.

[5] M. Tatari, M. Fard, N. Nasrollahzadeh, M. Mahjoob, "CAE characterization and optimization of automotive seat rattle noise," World Journal of Engineering and Technology, vol. 2, pp. 201-210, 2014.

[6] S. M. Kuo, H. T. Wu, "Nonlinear adaptive bilinear filters for active noise control systems," Circuits and Systems I: Regular Papers, IEEE Transactions on, vol. 52, pp. 617-624, 2005.

[7] S. D. Snyder, N. Tanaka, "Active control of vibration using a neural network," Neural Networks, IEEE Transactions on, vol. 6, pp. 819-828, 1995.

[8] N. V. George, G. Panda, "Advances in active noise control: A survey, with emphasis on recent nonlinear techniques," Signal Processing, vol. 93, pp. 363-377, 2013.

[9] M. Tokhi, R. Wood, "Active noise control using radial basis function networks," Control Engineering Practice, vol. 5, pp. 1311-1322, 1997.

[10] K. T. Chen, C. H. Chou, S. H. Chang, Y. H. Liu, "Adaptive fuzzy neural network control on the acoustic field in a duct," Applied Acoustics, vol. 69, pp. 558-565, 2008.

[11] H. Zhao, X. Zeng, Z. He, T. Li, W. Jin, "Nonlinear adaptive filter-based simplified bilinear model for multichannel active control of nonlinear noise processes," Applied Acoustics, vol. 74, pp. 1414-1421, 2013.

[12] O. Asgari Gashteroodkhani, B. Vahidi, A. Zaboli, "Time-time matrix z-score vector-based fault analysis method for series-compensated transmission lines," Turkish Journal of Electrical Engineering & Computer Sciences, vol. 25, pp. 2647-2659, 2017.

[13] N. V. George, G. Panda, "Active control of nonlinear noise processes using cascaded adaptive nonlinear filter," Applied Acoustics, vol. 74, pp. 217-222, 2013.

[14] N. V. George, A. Gonzalez, "Convex combination of nonlinear adaptive filters for active noise control," Applied Acoustics, vol. 76, pp. 157-161, 2014.

[15] S. Dashti, Z. Alzahra, M. Jafari, M. Gholami, M. A. Shoorehdeli, "Neural adaptive control based on backstepping and feedback linearization for electro hydraulic servo system," in Intelligent Systems (ICIS), 2014 Iranian Conference on. Bam: IEEE, pp. 1–6, 2014.

[16] Z. A. S. Dashti, Zohreh, M. Jafari, M. Gholami, M. A. Shoorehdeli, M. Teshnehlab, "Speed control of a Digital Servo System using parallel distributed compensation controller and Neural Adaptive controller," 13th Iranian Conference on Fuzzy Systems (IFSC), pp. 1-6, 27-29 Aug. 2013.

[17] Z. A. S. Dashti, M. Gholami, M. Jafari, M. A. Shoorehdeli, Hajimani, "Neural Adaptive controller for Magnetic levitation system", Iranian Conference on Intelligent Systems, Bam, Iran, 2014.

[18] C.-Y. Chang, "Simple approaches to improve the performance of noise cancellation," Journal of Vibration and Control, vol. 15, pp. 1875-1883, 2009.

[19] M. Akraminia, M. J. Mahjoob, M. Tatari, "Active noise control using adaptive POLYnomial Gaussian WinOwed wavelet networks," Journal of Vibration and Control, vol. 21, pp. 3020-3033 January 28, 2014.

[20] M. Akraminia, M. Mahjoob, M. Ghadami, "Nonlinear active noise control using adaptive rational functions with second-order poles frames: Simulation and experimental results," Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, vol. 229, pp. 1708-1723, 2015.

[21] M. Akraminia, M. J. Mahjoob, A. H. Niazi, "Feedforward active noise control using wavelet frames: simulation and experimental results," Journal of Vibration and Control, vol. 23, pp. 555-573, 2017.

[22] M. Akraminia, M. J. Mahjoob, "Adaptive feedback active noise control using wavelet frames: simulation and experimental results," Journal of Vibration and Control, vol. 22, pp. 1895-1912, 2016.

[23] S. Mallat, A wavelet tour of signal processing: Academic press, 1999.

[24] I. Daubechies, Ten lectures on wavelets vol. 61: SIAM, 1992.

[25] G. Lekutai, "Adaptive self-tuning neuro wavelet network controllers," Virginia Polytechnic Institute and State University, 1997.

[26] S. A. Imhoff, D. Y. Roeum, M. R. Rosiek, "New classes of frame wavelets for applications," in SPIE's 1995 Symposium on OE/Aerospace Sensing and Dual Use Photonics, 1995, pp. 923-934.

[27] Y. L. Zhou, Q. Z. Zhang, X. D. Li, W. S. Gan, "Analysis and DSP implementation of an ANC system using a filtered-error neural network," Journal of Sound and Vibration, vol. 285, pp. 1-25, 2005.