Applied Acoustics 132 (2018) 19–25

Contents lists available at ScienceDirect

Applied Acoustics

journal homepage: www.elsevier.com/locate/apacoust

Technical note

A bilinear functional link artificial neural network filter for nonlinear active noise control and its stability condition

Dinh Cong Le, Jiashu Zhang*, Yanjie Pang

Sichuan Province Key Lab of Signal and Information Processing, Southwest Jiaotong University, Chengdu 610031, PR China

ARTICLE INFO

Keywords:
Nonlinear active noise control
FLANN
Generalized FLANN
Bilinear filter

ABSTRACT

Since the functional link artificial neural network (FLANN) filter using trigonometric expansions do not exploit cross-terms (products of input samples and /or past output samples with different time shifts), its performance for nonlinear active noise control (ANC) can be considerably degraded, especially in strong nonlinearity environment. In order to overcome this drawback, a novel bilinear FLANN (BFLANN) filter for the nonlinear ANC is proposed in this paper. In addition, a sufficient condition that guarantees the stability of the BFLANN filter is also presented. Simulation results demonstrate that the proposed BFLANN filter based nonlinear ANC can achieve better performance than the FLANN and generalized FLANN (GFLANN) filters based nonlinear ANC in the presence of strong nonlinearity.

1. Introduction

In order to solve acoustic noise problems (such as noises from engines, fans and compressors…), the active noise control (ANC) systems have become a potential solution and attracted attention of many researchers [1,2]. It is well known that conventional ANC system using the linear filter as a linear controller has been widely and popularly applied [3,4]. However, since the linear ANC system does not take account of nonlinearity that is contained in the components of the actual ANC systems, its performance may be seriously degraded [5–7].

To deal with this problem, various types of nonlinear filters based on the polynomial filter (PF) and neural networks (NNs) such as radial basic function (RBF) [6], multilayer neural network (MLNN) [8–10], recurrent neural network (RNN) [11] and Volterra filters (VFs) [12,13] have been used in the nonlinear ANC systems with good result. These nonlinear filters, however, exhibit numerous disadvantages such as the complicated architecture and the heavy computational burden of its implementation.

In addition, an effective alternative to nonlinear filter in nonlinear ANC systems is the well-known FLANN filter using trigonometric expansions[14–20], and its several modifications (such as recursive FLANN [21,22], reduced feedback FLANN [23] and hybrid active noise control system-based FLANN [24–27]) have also been presented in recent years.

However, as pointed out in [28], the performance of nonlinear ANC systems using the FLANN can be negatively affected because of the lack of cross-terms. This becomes more serious as the strong nonlinearity is present in the components of the actual ANC systems. In [28], Sicuranza and Carini have proposed a GFLANN filter with the use of trigonometric expansions including suitable cross-terms for nonlinear ANC system. Research results indicate that GFLANN controller has better performance than FLANN and high-order Volterra controllers in the presence of strong nonlinearity in the primary or secondary paths. With the aim of further exploiting the advantage of the cross product terms, a bilinear FLANN (BFLANN) filter for nonlinear ANC system is presented in this paper. Unlike the GFLANN, thanks for employing both feedback and feedforward polynomials, the proposed filter can accurately model nonlinear systems with shorter filter length.

The rest of this paper is organized as follows: Section 2 proposes new BFLANN filter for nonlinear ANC system; Section 3 and 4 present the analysis of stability condition and computational complexity, respectively; Section 5 provides computer simulation studies of the proposed controller; finally, the conclusion is drawn in Section 6.

2. BFLANN filter for nonlinear ANC system

The block diagram of the NANC system using BFLANN filter is illustrated in Fig. 1. Here, the transfer functions P(z), S(z) represent the primary path from the reference microphone to the error microphone and the secondary path from the output of the filter to the output of the error microphone, respectively.

The $P(z)$, $S(z)$ and input signal $x(n)$ may contain nonlinearities.

In order to avoid the confusion and complication, in this section we just introduce the BFLANN of order $P = 1$. The approach can be easily

* Corresponding author.
E-mail address: jszhang@home.swjtu.edu.cn (J. Zhang).

https://doi.org/10.1016/j.apacoust.2017.10.023
Received 3 March 2017; Received in revised form 6 October 2017; Accepted 19 October 2017

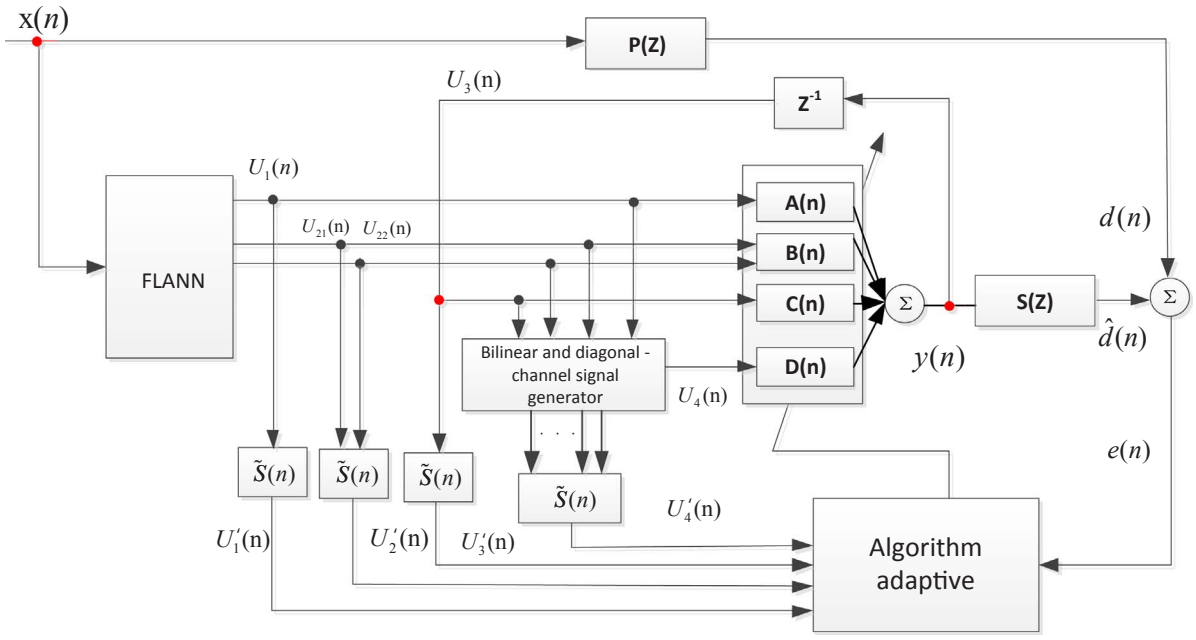

Fig. 1. The nonlinear ANC system based on the BFLANN filter.

extended to a BFLANN filter of any order P.

Therefore, similar to [29], the relationship between input $x(n)$ and output $y(n)$ of the BFLANN filter with a memory length of L expressed as

$$\begin{aligned} y(n) &= \sum_{j=0}^{L-1} a_j(n)x(n-j) + \sum_{j=0}^{L-1} b1_j(n)\sin(\pi x(n-j)) \\ &+ \sum_{j=0}^{L-1} b2_j(n)\cos(\pi x(n-j)) + \sum_{j=1}^{L-1} c_j(n)y(n-j) \\ &+ \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d1_{i,j}(n)x(n-i)y(n-j) + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d2_{i,j}(n)\sin(\pi x(n \\ &-i))y(n-j) + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d3_{i,j}(n)\cos(\pi x(n-i))y(n-j) \end{aligned} \tag{1}$$

where $a_j(n)$, $b1_j(n)$ and $b2_j(n)$ are feedforward coefficients extended by FLANN with the order $P = 1$; $c_j(n)$ are feedback coefficients; $d1_{i,j}(n)$, $d2_{i,j}(n)$ and $d3_{i,j}(n)$ are the coefficients of cross-terms.

Similar to [30–33], in order to achieve an efficient implementation based on a filter bank formed with FIR filters, (1) can be equivalently written as follows

$$\begin{aligned} y(n) &= \sum_{j=0}^{L-1} a_j(n)x(n-j) + \sum_{j=0}^{L-1} b1_j(n)\sin(\pi x(n-j)) \\ &+ \sum_{j=0}^{L-1} b2_j(n)\cos(\pi x(n-j)) + \sum_{j=1}^{L-1} c_j(n)y(n-j) \\ &+ \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} g1_{i,j}(n)x(n-j)y(n-j-i) + \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} h1_{i,j}(n)x(n-i \\ &-j)y(n-1-j) + \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} g2_{i,j}(n)sin(\pi x(n-j))y(n-j-i) \\ &+ \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} h2_{i,j}(n)sin(\pi x(n-i-j))y(n-1-j) \\ &+ \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} g3_{i,j}(n)cos(\pi x(n-j))y(n-j-i) \\ &+ \sum_{i=1}^{L-1}\sum_{j=0}^{L-i-1} h3_{i,j}(n)cos(\pi x(n-i-j))y(n-1-j) \end{aligned} \tag{2}$$

where i denotes the bilinear filter channel number and j designates the time index; $g1_{i,j}(n)$, $h1_{i,j}(n)$, $g2_{i,j}(n)$, $h2_{i,j}(n)$ and $g3_{i,j}(n)$, $h3_{i,j}(n)$ are the coefficients of the cross-terms.

To derive the adaptive algorithm for BFLANN filter, the model in (2) is rewritten under the vector form as follows

$$\begin{aligned} y(n) &= A^T(n)U_1(n) + B_1^T(n)U_{21}(n) + B_2^T(n)U_{22}(n) + C^T(n)U_3(n) \\ &+ \sum_{i=1}^{L-1} G1_i^T(n)V1_i(n) + \sum_{i=1}^{L-1} H1_i^T(n)Q1_i(n) + \sum_{i=1}^{L-1} G2_i^T(n)V2_i(n) \\ &+ \sum_{i=1}^{L-1} H2_i^T(n)Q2_i(n) + \sum_{i=1}^{L-1} G3_i^T(n)V3_i(n) + \sum_{i=1}^{L-1} H3_i^T(n)Q3_i(n) \end{aligned} \tag{3}$$

where signal vectors and their corresponding coefficient vectors are listed below

$$A(n) = [a_0(n)a_1(n)\cdots a_{L-1}(n)]^T \tag{4}$$

$$U_1(n) = [x(n)x(n-1)\cdots x(n-L+1)]^T \tag{5}$$

$$B_1(n) = [b1_0(n)b1_1(n)\cdots b1_{L-1}(n)]^T \tag{6}$$

$$\begin{aligned} &U_{21}(n) \\ &= [\sin(\pi x(n))\sin(\pi x(n-1))\cdots\sin(\pi x(n-L+2))\sin(\pi x(n-L+1))]^T \end{aligned} \tag{7}$$

$$B_2(n) = [b2_0(n)b2_1(n)\cdots b2_{L-1}(n)]^T \tag{8}$$

$$\begin{aligned} &U_{22}(n) \\ &= [cos(\pi x(n))cos(\pi x(n-1))\cdots\cos(\pi x(n-L+2))cos(\pi x(n-L+1))]^T \end{aligned} \tag{9}$$

$$C(n) = [c_1(n)c_2(n)\cdots c_{L-1}(n)]^T \tag{10}$$

$$U_3(n) = [y(n-1)y(n-2)\cdots y(n-L+1)]^T \tag{11}$$

For $i = 1,2,\ldots,L-1$

$$G1_i^T(n) = [g1_{i,0}(n)g1_{i,1}(n)\cdots g1_{i,L-i-1}(n)]^T \tag{12}$$

$$V1_i^T(n) = [x(n)y(n-i)x(n-1)y(n-i-1)\ldots x(n-L+1+i)y(n-L+1)]^T \tag{13}$$

$$G2_i^T(n) = [g2_{i,0}(n)g2_{i,1}(n)\cdots g2_{i,L-i-1}(n)]^T \tag{14}$$

$$V2_i^T(n) = [\sin(\pi x(n))y(n-i)\sin(\pi x(n-1))y(n-i-1)\ldots \sin(\pi x(n-L+1+i))y(n-L+1)]^T \tag{15}$$

$$G3_i^T(n) = [g3_{i,0}(n)g3_{i,1}(n)\cdots g3_{i,L-i-1}(n)]^T \tag{16}$$

$$V3_i^T(n) = [cos(\pi x(n))y(n-i)cos(\pi x(n-1))y(n-i-1)\ldots \cos(\pi x(n-L+1+i))y(n-L+1)]^T \tag{17}$$

$$H1_i^T(n) = [h1_{i,0}(n)h1_{i,1}(n)\cdots h1_{i,L-i-1}(n)]^T \tag{18}$$

$$Q1_i^T(n) = [x(n-i)y(n-1)x(n-i-1)y(n-2)\ldots x(n-L+1)y(n-L+i)]^T \tag{19}$$

$$H2_i^T(n) = [h2_{i,0}(n)h2_{i,1}(n)\cdots h2_{i,L-i-1}(n)]^T \tag{20}$$

$$Q2_i^T(n) = [\sin(\pi x(n-i))y(n-1)\sin(\pi x(n-i-1))y(n-2)\ldots \sin(\pi x(n-L+1))y(n-L+i)]^T \tag{21}$$

$$H3_i^T(n) = [h3_{i,0}(n)h3_{i,1}(n)\cdots h3_{i,L-i-1}(n)]^T \tag{22}$$

$$Q3_i^T(n) = [cos(\pi x(n-i))y(n-1)cos(\pi x(n-i-1))y(n-2)\ldots \cos(\pi x(n-L+1))y(n-L+i)]^T \tag{23}$$

By combination of (4), (6), (8), (10), (12), (14), (16), (18), (20) and (22), we get the coefficient vector as an overall vector $W(n)$ as follows

$$W(n) = [A^T(n)B_1^T(n)B_2^T(n)C^T(n)G1_1^T(n)\ldots G1_{L-1}^T(n)G2_1^T(n)\ldots G2_{L-1}^T(n)G3_1^T(n)\ldots G3_{L-1}^T(n)H1_1^T(n)\ldots H1_{L-1}^T(n)H2_1^T(n)\ldots H2_{L-1}^T(n)H3_1^T(n)\ldots H3_{L-1}^T(n)]^T \tag{24}$$

Similarly, we can combine (5), (7), (9), (11), (13), (15), (17), (19), (21) and (23), to generalize signal vector $U(n)$ as follow

$$U(n) = [U_1^T(n)U_{21}^T(n)U_{22}^T(n)U_3^T(n)V1_1^T(n)\ldots V1_{L-1}^T(n)V2_1^T(n)\ldots V2_{L-1}^T(n)V3_1^T(n)\ldots V3_{L-1}^T(n)Q1_1^T(n)\ldots Q1_{L-1}^T(n)Q2_1^T(n)\ldots Q2_{L-1}^T(n)Q3_1^T(n)\ldots Q3_{L-1}^T(n)]^T \tag{25}$$

With the definitions in (24) and (25), the BFLANN filter output can be simplified to

$$y(n) = W^T(n)U(n) \tag{26}$$

Fig. 2 illustrates the diagonal-channels of $V2_i(n)$, $Q2_i(n)$ vectors for the case $L = 4$

The residual noise sensed by the error microphone, which can is given by (see Fig. 1)

$$e(n) = d(n) - \hat{d}(n) \tag{27}$$

where $d(n)$ is primary noise signal at the cancellation point and $\hat{d}(n)$ is the signal generated by the adaptive control and propagated through the secondary path to the cancellation point.Similar to [34], we use a virtual secondary path concept $\widetilde{S}(n)$ to achieve a unified filtered-x

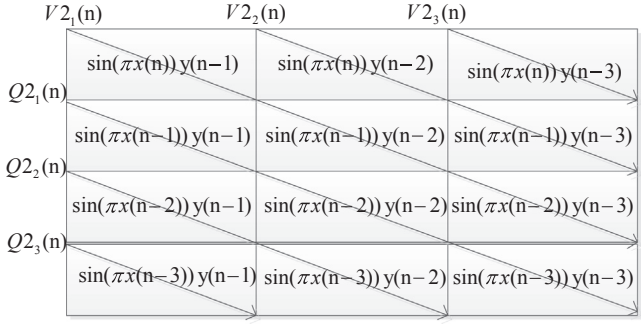

Fig. 2. Illustrate the cross-terms of $V2_i(n)$, $Q2_i(n)$ vectors for the case $L = 4$.

structure for both NANC/linear secondary path (LSP) and NANC/non-linear secondary path (NSP). It is defined as a time-varying filter with coefficients as follows

$$\widetilde{S}(n) = [g(n,0)g(n,1)\ldots g(n,M)]^T = \left[\frac{\partial\hat{d}(n)}{\partial y(n)}\frac{\partial\hat{d}(n)}{\partial y(n-1)}\ldots\frac{\partial\hat{d}(n)}{\partial y(n-M)}\right]^T \tag{28}$$

where M is the length of the virtual secondary path.

Using the same approach in [29] and assumptions in [35], the coefficient update equations of the BFLANN filter are derived and summarized as

$$A(n+1) = A(n) + \frac{\mu_a}{\|U'(n)\|^2}U_1'(n)e(n) \tag{29a}$$

$$B(n+1) = B(n) + \frac{\mu_b}{\|U'(n)\|^2}U_2'(n)e(n) \tag{29b}$$

$$C(n+1) = C(n) + \frac{\mu_c}{\|U'(n)\|^2}U_3'(n)e(n) \tag{29c}$$

$$D(n+1) = D(n) + \frac{\mu_d}{\|U'(n)\|^2}U_4'(n)e(n) \tag{29d}$$

where

$$U_1'(n) = \widetilde{S}(n) * U_1(n) \tag{30}$$

$$B(n) = [B_1(n)^T B_2(n)^T]^T \tag{31}$$

$$U_2'(n) = [[\widetilde{S}(n) * U_{21}(n)]^T[\widetilde{S}(n) * U_{22}(n)]^T]^T \tag{32}$$

$$U_3'(n) = \widetilde{S}(n) * U_3(n) \tag{33}$$

$$D(n) = [G1_1^T(n)\ldots G1_{L-1}^T(n)G2_1^T(n)\ldots G2_{L-1}^T(n)G3_1^T(n)\ldots G3_{L-1}^T(n)H1_1^T(n)\ldots H1_{L-1}^T(n)H2_1^T(n)\ldots H2_{L-1}^T(n)H3_1^T(n)\ldots H3_{L-1}^T(n)]^T \tag{34}$$

$$U_4'(n) = \{[\widetilde{S}(n) * V1_1(n)]^T\ldots[\widetilde{S}(n) * V1_{L-1}(n)]^T[\widetilde{S}(n) * V2_1(n)]^T\ldots [\widetilde{S}(n) * V2_{L-1}(n)]^T[\widetilde{S}(n) * V3_1(n)]^T\ldots [\widetilde{S}(n) * V3_{L-1}(n)]^T[\widetilde{S}(n) * Q1_1(n)]^T\ldots [\widetilde{S}(n) * Q1_{L-1}(n)]^T[\widetilde{S}(n) * Q2_1(n)]^T\ldots [\widetilde{S}(n) * Q2_{L-1}(n)]^T[\widetilde{S}(n) * Q3_1(n)]^T\ldots[\widetilde{S}(n) * Q3_{L-1}(n)]^T\}^T \tag{35}$$

$$U'(n) = \left[U_1'^T(n)U_2'^T(n)U_3'^T(n)U_4'^T(n)\right]^T \tag{36}$$

where $*$ denotes convolution and $U_1'(n)$, $U_2'(n)$, $U_3'(n)$, $U_4'(n)$ are the filtered version of $U_1(n)[U_{21}^T(n)U_{22}^T(n)]^T U_3(n),[V1_1^T(n)\ldots V1_{L-1}^T(n)V2_1^T(n)\ldots V2_{L-1}^T(n)V3_1^T(n)\ldots Q1_1^T(n)\ldots Q1_{L-1}^T(n)Q2_1^T(n)\ldots Q2_{L-1}^T(n)Q3_1^T(n)\ldots Q3_{L-1}^T(n)]^T$ by the virtual secondary path $\widetilde{S}(n)$, respectively.$\|U(n)'\|^2 = U'^T(n)U'(n)$ is the squared Euclidean norm which is used as a normalization factor. μ_a, μ_b, μ_c and μ_d are the learning rate for the feed-forward coefficients $A(n)$, $B(n)$, the feedback coefficients $C(n)$, and the cross-term coefficients $D(n)$, respectively.

3. The stability condition

The same as in the IIR filter, most bilinear systems are inherently unstable. Consequently, in this section we propose a sufficient condition to guarantee the output of BFLANN system which is bounded whenever its input signal is bounded by finite constant.

Using time-invariant operator, the model in (1) can be rewritten as

$$(1-C(z))y(n) = A(z)x(n) + B1(z)\sin(\pi x(n)) + B2(z)\cos(\pi x(n)) + D1(z)[x(n),y(n)] + D2(z)[\sin\pi x(n),y(n)] + D3(z)[\cos\pi x(n),y(n)] \quad (37)$$

where

$$A(z) = \sum_{j=0}^{L-1} a_j z^{-j}; B1(z) = \sum_{j=0}^{L-1} b1_j z^{-j}; B2(z) = \sum_{j=0}^{L-1} b2_j z^{-j}$$

$$C(z) = \sum_{j=1}^{L-1} c_j z^{-j}; D1(z) = \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d1_{ij}[z^{-i},z^{-j}]$$

$$D2(z) = \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d2_{ij}[z^{-i},z^{-j}]; D3(z) = \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} d3_{ij}[z^{-i},z^{-j}]$$

$z^{-j}x(n) = x(n-j); z^{-j}\sin\pi x(n) = \sin(\pi x(n-j));$
$z^{-j}\cos\pi x(n) = \cos(\pi x(n-j)); z^{-j}y(n) = y(n-j);$
$[z^{-i},z^{-j}][x(n),y(n)] = x(n-i)y(n-j);$
$[z^{-i},z^{-j}][\sin(\pi x(n)),y(n)] = \sin(\pi x(n-i))y(n-j);$
$[z^{-i},z^{-j}][\cos(\pi x(n)),y(n)] = \cos(\pi x(n-i))y(n-j)$

By utilizing the approach in [36], a sufficient condition for the stability of the BFLANN filter is given by the following theorem.

Theorem. *Let us call $p_1, p_2, \ldots, p_{L-1}$ denote the zeros of the polynomial $z^{L-1}(1\text{-}C(z))$ and the input signal is bounded by $M_x > 0$ for every n, then a sufficient condition to produce a bound output y(n) is shown as follows*

$$\begin{cases} |p_i| < 1 for \quad i = 1,2,\ldots,L-1 \\ \Phi\left\{M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right\} < 1 \end{cases} \quad (38)$$

where

$$\Phi = \frac{1}{\prod_{i=1}^{L-1}(1-|p_i|)} \quad (39)$$

and y(n) is bounded by

$$|y(n)| \leqslant \frac{\Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right)}{1-\Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right)} \quad (40)$$

Proof. first of all, we assume that model expressed in (37) is initially at rest, it then is defined as follows

$$\begin{cases} (1-C(z))y_1(n) = A(z)x(n) + B1(z)\sin(\pi x(n) + B2(z)\cos(\pi x(n) \\ (1-C(z))y_k(n) = D1(z)[x(n),y_{k-1}(n)] + D2(z)[\sin(\pi x(n)), \\ y_{k-1}(n)] + D3(z)[\cos(\pi x(n)),y_{k-1}(n)] \\ for \quad k = 2,3,\ldots,n+1 \\ y_1(T) = y_2(T) = \ldots = y_{n+1}(T) = 0 \quad for \quad T < -1 \end{cases} \quad (41)$$

Similar to [36], we can deduce as

$$y_1(n) = \left(\prod_{i=1}^{L-1}\left\{\sum_{l=0}^{n} (p_i)^l z^{-l}\right\}\right)[A(z)x(n) + B1(z)\sin(\pi x(n) + B2(z)\cos(\pi x(n)] \quad (42)$$

and

$$y_k(n) = \left\{\prod_{i=1}^{L-1}\left\{\sum_{l=0}^{n} (p_i)^l z^{-l}\right\}\right\}[D1(z)[x(n),y_{k-1}(n)] + D2(z)[\sin(\pi x(n)),y_{k-1}(n)] + D3(z)[\cos(\pi x(n)),y_{k-1}(n)] \quad (43)$$

for k = 2,3,…,n + 1.For x(n) is bounded by $M_x > 0$ for every n, the sine and cosine functions are bounded by unity. On the other hand, since $|p_i| < 1$, the following equality hold

$$\sum_{l=0}^{\infty} |p_i|^l = (1-p_i)^{-1} \quad (44)$$

By substituting (44) in (42) and using the above conditions, we have

$$|y_1(n)| \leqslant \Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right) \quad (45)$$

From (45) it results

$$D1(z)[x(n),y_1(n)] + D2(z)[\sin(\pi x(n)),y_1(n)] + D3(z)[\cos(\pi x(n)),y_1(n)] \leqslant \left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right) \times \left(\Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right)\right) \quad (46)$$

Now, we easily derive that $y_2(n)$ is bounded by

$$|y_2(n)| \leqslant \Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right) \times \left(\Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right)\right) \quad (47)$$

Exploiting the induction method, we can go to

$$|y_k(n)| \leqslant \left(\Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right)\right)^{k-1} \times \left(\Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right)\right) \quad (48)$$

According to the approach in [36], we easily show that $y(n) = \sum_{k=1}^{n+1} y_k(n)$ is the unique solution to the (37). Therefore, we have

$$|y(n)| \leqslant \sum_{k=1}^{n+1} |y_k(n)| \leqslant \frac{1-\left(\Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right)\right)^{n+1}}{1-\Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right)} \times \left(\Phi(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|)\right) \quad (49)$$

Moreover, since

$$\Phi\left\{M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right\} < 1$$

It is easy to see that

$$|y(n)| \leqslant \frac{\Phi\left(M_x \sum_{j=0}^{L-1} |a_j| + \sum_{j=0}^{L-1} |b1_j| + \sum_{j=0}^{L-1} |b2_j|\right)}{1-\Phi\left(M_x \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d1_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d2_{i,j}| + \sum_{i=0}^{L-1}\sum_{j=1}^{L-1} |d3_{i,j}|\right)}$$

This completes the proof.

4. Computational complexity analysis

To compare the computational complexity of proposed BFLANN controller with FLANN and GFLANN controllers, we carried out the analysis of their computational complexity in terms of the number of multiplications and additions for both the linear secondary path (LSP) and the nonlinear secondary path (NSP). Assume that L is the memory length of the BFLANN; N is the memory length and N_d denotes the number of diagonals involving cross-terms with delay difference of the GFLANN; K and P are the memory length and the order of expansion function of the FLANN; M is the memory size of the virtual secondary path.

4.1. Computational complexity for NANC/LSP

The NANC system using the adaptive BFLANN filter requires major operations as follows.

The operations for generating the cross-terms require $6(L-1)$ multiplications.

The operations for generating the adaptive bilinear FLANN filter output, which require $3L + (L-1) + 3L(L-1)$ multiplications and $3L + (L-1) + 3L(L-1) - 1$ additions.

The operations for generating the filtered signals through pass of the virtual secondary path, which requires $(6(L-1)+4))M$ multiplications and $(6(L-1)+4))(M-1)$ additions.

The operations for updating filter coefficients defined in (29a–29d), which requires $2(3L + (L-1) + 3L(L-1)) + 4$ multiplications and $2(3L + (L-1) + 3L(L-1)) - 1$ additions.

Total computational load of BFLANN is $9L^2 + 9L + 6LM - 2M - 5$ multiplications and $9L^2 - 3L + 6LM - 2M - 3$ additions.

The same analysis we obtain total computational load of GFLANN and FLANN filters are $(3 + 2N_d)M + 3N^2 + 6N + 2N_d + 3$ multiplications, $3N^2 + 6N + 2N_dM + 3M - 2N_d - 5$ additions and $3(K + 2PK) + 2 + (2P + 1)M$ multiplications, $3(K + 2PK) - 2 + (2P + 1)(M - 1)$ additions, respectively. (In these analyses, all adaptive filters are updated by Filtered-X NLMS-based algorithms).

4.2. Computational complexity for NANC/NSP

When the secondary path is nonlinear, the virtual secondary filter is time-varying. Consequently, we cannot take advantage of the delay relationship existing in the nonlinear states. Thus, each element of $U(n)$ must be filtered by the virtual secondary path $\widetilde{S}(n)$.

By calculating similar to the case of NANC/LSP, we obtain the total computational load of the filters as follows:

- For the FLANN requires $(3 + M)(K + 2PK) + 2$ multiplication and $(2 + M)(K + 2PK) - 2$ addition.
- For the GLANN requires $(3 + M)(N^2 + 2N) + 2N_d + 3$ multiplication and $(M + 2)(N^2 + 2N) - 2$ addition.
- For the BLANN requires $(3L^2 + L - 1)(M + 3) + 6(L - 1) + 4$ multiplication and $(3L^2 + L - 1)(M + 2) - 2$ addition.

5. Simulation

In order to prove the effectiveness of our proposed filter for the NANC system, several simulation results are provided in this section. The performance of the BFLANN filter is compared with the FLANN and GFLANN filters for both NANC/LSP and NANC/NSP.

In the experiments, the performance of the different filters will be measured in terms of the normalized mean-square error (NMSE) which is obtained by averaging over 200 independent runs.

$$NMSE = 10\log10\left(\frac{E(e^2(n))}{\delta_d^2}\right) \tag{50}$$

where δ_d^2 is the variance of the primary noise at the cancellation point, $e^2(n)$ is the square of the error at nth iteration and $E(.)$ is the expectation operator.

The memory length of the FLANN and GFLANN filters are chosen as $N = K = 10$; and of the BFLANN denoted as $L = 6$. The function expansion of the input signal is third-order type ($P = 3$) for the FLANN and first-order type ($P = 1$) for GFLANN and BFLANN. The parameter for expanding nonlinear function of GFLANN is chosen as $N_d = 9$.

The ensemble curves are smoothed with a rectangular window of length equal to 20 samples in order to better discern the curves behavior.

5.1. Experiment I

In this experiment, we simulate NANC system with the nonlinear secondary path and the primary path exhibits high nonlinear behavior. Here, the primary path $P(z)$ is modeled by a Volterra series whose input $x(n)$ and output $d(n)$ relationship is described as

$$\begin{aligned} d(n) = {} & x(n) + 0.8x(n-1) + 0.3x(n-2) + 0.4x(n-3) - 0.8x(n)x(n-1) \\ & + 0.9x(n)x(n-2) + 0.7x(n) \times x(n-3) - 3.9x^2(n-1)x(n-2) \\ & - 2.6x^2(n-1) \times x(n-3) + 2.1x^2(n-2)x(n-3) \end{aligned} \tag{51}$$

and the secondary path has the input $y(n)$ to output $\hat{d}(n)$ relationship

$$\begin{aligned} \hat{d}(n) = {} & y(n) + 0.35y(n-1) + 0.09y(n-2) - 0.5y(n)y(n-1) \\ & + 0.4y(n)y(n-2) \end{aligned} \tag{52}$$

The reference signal is white Gaussian noise. The learning rate of FLANN is set to $\eta_H = 0.01$, $\eta_w = 0.9$ for the nonlinear and linear parts, respectively. Learning rate of GFLANN are $\eta_W = 0.3$, $\eta_H = 0.05$ and $\eta_c = 0.5$ for the linear, the sin(.) cos(.) functions and the cross-terms parts, respectively. The learning rate of BFLANN $\mu_a = 0.7$, $\mu_b = 0.02$ for the feedforward coefficients $A(n)$, $B(n)$, respectively, $\mu_c = 0.01$ and $\mu_d = 0.6$ for the feedback coefficient $C(n)$ and the cross-term coefficients $D(n)$, respectively.

Fig. 3 illustrates the averaged NMSE performance curves of the filters for the reference signal is white Gaussian noise. It clearly indicates that the BFLANN exhibits better performance as compared to FLANN and GFLANN.

5.2. Experiment II

In practical NANC system, the secondary path can be seen as the nonlinear effect of the power amplifier and loudspeaker at the output of the filter, and thus it can be modeled by a Hammerstein filter with a memoryless nonlinearity and the linear filter as

$$\begin{aligned} v(n) &= \tanh(y(n)) \\ \hat{d}(n) &= v(n) + 0.2v(n-1) + 0.05v(n-2) \end{aligned} \tag{53}$$

Here, the input signal is white Gaussian noise. The learning rate of FLANN is set to $\eta_H = 0.01$; $\eta_w = 0.8$ for the nonlinear and linear parts, respectively. The learning rate of GFLANN are set to $\eta_W = 0.2$, $\eta_H = 0.05$ and $\eta_c = 0.6$ for the linear, the sin(.) cos(.) functions and the cross-terms parts, respectively. The learning rate of BFLANN is selected as: $\mu_a = 0.6$, $\mu_b = 0.01$, $\mu_c = 0.01$ and $\mu_d = 0.7$.

Table 1 illustrates the computational complexity of the controllers for NANC/NSP.

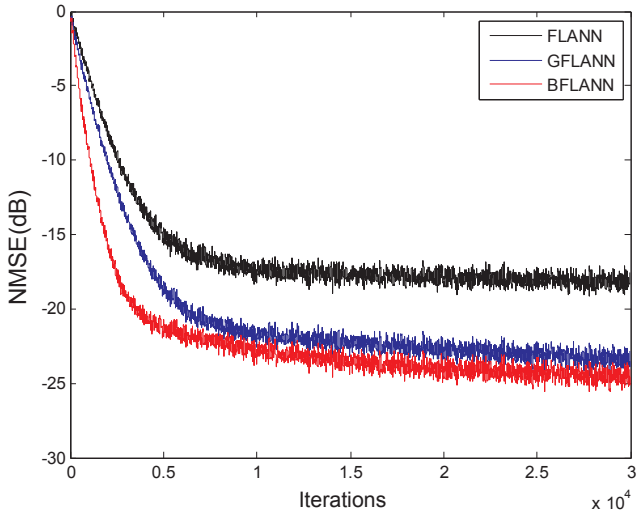

Fig. 3. Learning curves of different filters for NANC/NSP, the secondary path is the Volterra model.

Table 1
Computational requirements of the controllers for NANC/NSP.

Nonlinear controller		Multiplications	Additions
FLANN	(K = 10,P = 3,M = 3)	422	348
GFLANN	(N = 10,N_d = 9,M = 3)	741	598
BFLANN	(L = 6,M = 3)	712	563

Fig. 4 shows a comparative plot of the NMSE achieved by NANC with the BFLANN, GFLANN and FLANN filters versus the number of iterations. It is observed that the performance of the BFLANN and GFLAN in NANC systems is superior to the FLANN for nonlinear secondary path, while the BFLANN is slightly better than the GFLANN. In addition, Table 1 also shows that in the case of NANC/NSP, the computational complexity of BFLANN is lower than that of GFLANN.

5.3. Experiment III

In this experiment, a logistic chaotic noise is chosen as the reference noise source as in [5], which is generated using the recursive equation as follows

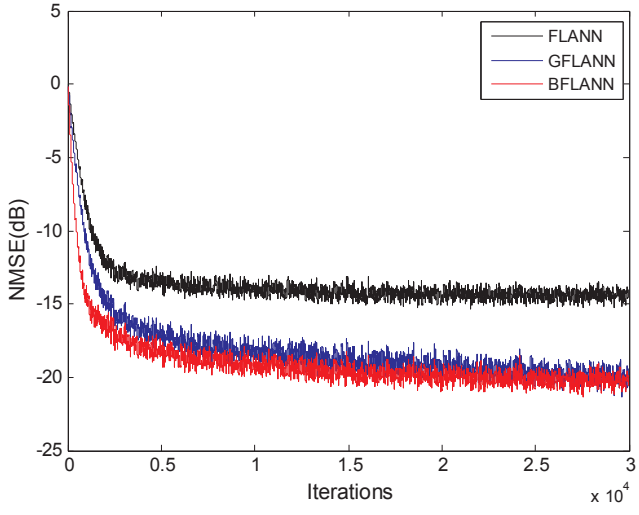

Fig. 4. Learning curves of different filters for NANC/NSP, the secondary path is the Hammerstein model.

Table 2
Computational requirements of the controllers for NANC/LSP.

Nonlinear controller		Multiplications	Additions
FLANN	(K = 10,P = 3,M = 5)	247	236
GFLANN	(N = 10,N_d = 9,M = 5)	483	442
BFLANN	(L = 6,M = 5)	543	473

$$x(n+1) = \lambda x(n)[1-x(n)] \tag{54}$$

where $\lambda = 4$ and $x(0) = 0.9$ are selected. This noise process is normalized to have unit signal power.

The linear primary path is modeled as an FIR filter with transfer function

$$P(z) = z^{-8} - 0.3z^{-9} + 0.2z^{-10} \tag{55}$$

and the secondary path transfer function is taken as the non-minimum-phase model

$$S(z) = z^{-2} + 1.5z^{-3} - z^{-4} \tag{56}$$

The learning rate of FLANN is set to $\eta_H = \eta_W = 0.05$ for the nonlinear and linear parts. Learning rate of GFLANN are $\eta_W = 0.6$, $\eta_H = 0.05$ and $\eta_c = 0.01$ for the linear, the sin(.) cos(.) functions and the cross-terms parts, respectively. The learning rate of BFLANN is selected as: $\mu_a = 0.5$, $\mu_b = 0.2$, $\mu_c = 0.9$ and $\mu_d = 0.01$.

Table 2 illustrates the computational complexity of the controllers for NANC/LSP.

Fig. 5 illustrates the averaged NMSE performance curves of the filters using nonminimum-phase secondary path and chaotic reference signal. From Fig. 5 it is observed that the proposed BFLANN filter yields better NMSE performance compared to the FLANN and GFLANN filters. However, as shown in Table 2, the computational complexity of BFLANN is slightly higher than that of GFLANN.

5.4. Experiment IV

In this experiment, the primary noise at the cancellation point is generated based on the third-order polynomial model as given in [13]

$$d(n) = t(n-2) + 0.08t^2(n-2) - 0.04t^3(n-2) \tag{57}$$

where $t(n)$ is obtained by

$$t(n) = x(n) * f(n) \tag{58}$$

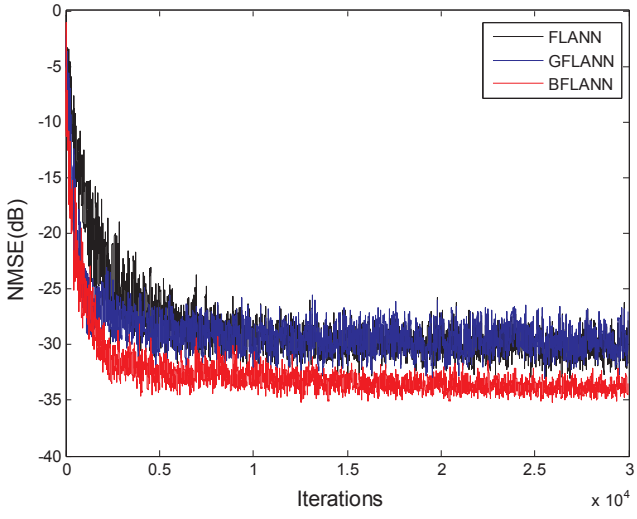

Fig. 5. Learning curves of different filters for NANC/LSP, the reference signal is a logistic chaotic noise, the secondary path is chosen as the non-minimum-phase model, the primary path is linear.

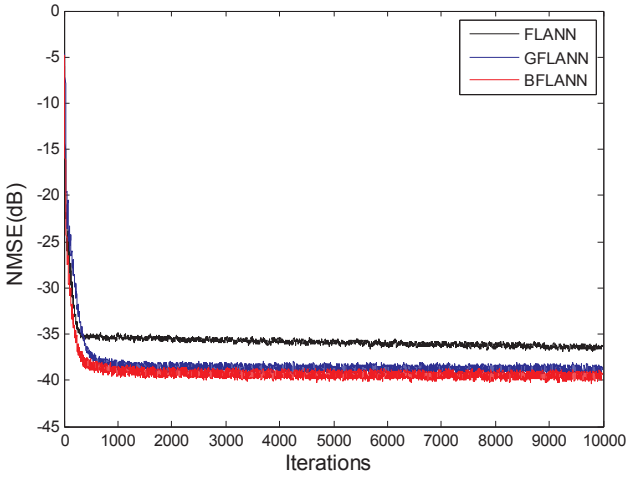

Fig. 6. Learning curves of different filters for NANC/LSP, the reference signal is a sinusoidal wav, the secondary path is chosen as the non-minimum-phase model, the primary path is nonlinear.

and $*$ denotes the convolution operation, $f(n)$ is the impulsive response of the transfer function as

$$f(z) = z^{-3} - 0.3z^{-4} + 0.2z^{-5} \tag{59}$$

The reference noise $x(n)$ is a sinusoidal wave of 500 Hz sampled at the rate of 8000 samples/s, which is obtained by

$$x(n) = \sqrt{2}\sin\left(\frac{2\pi 500n}{8000}\right) + \nu(n) \tag{60}$$

where $\nu(n)$ denotes a Gaussian noise of 40 dB SNR

The secondary path is chosen as the one used in experiment IIIThe learning rate of FLANN is set to $\eta_H = 0.4$, $\eta_w = 0.9$ for the nonlinear and linear parts, respectively. Learning rate of GFLANN are $\eta_W = 0.6$, $\eta_H = 0.35$ and $\eta_c = 0.1$ for the linear, the sin(.) cos(.) functions and the cross-terms parts, respectively. The learning rate of BFLANN are selected as: $\mu_a = 0.7$, $\mu_b = 0.1$, $\mu_c = 0.9$ and $\mu_d = 0.05$.

Fig. 6 depicts the performance comparison of NMSE for BFLANN, GFLANN and FLANN filters. It is clear that the proposed BFLANN filter achieves lower NMSE than that of GFLANN and FLANN filters.

In summary, simulation results and computational complexity analysis indicate that the FLANN requires the lowest computations, but its performance is also the lowest in comparison with GFLANN and BFLANN. The BFLANN requires more computations than GFLANN for the NANC/LSP case, but lower for the NANC/NSP case. However, the performance of the BFLANN is better than that of the GFLANN for both NANC/LSP and NANC/NSP.

6. Conclusion

In this paper, a novel adaptive BFLANN filter for the nonlinear ANC systems is proposed. Similar to the GFLANN, this filter also satisfies the properties of class of nonlinear filters whose output depends linearly on the filter coefficient and nonlinear expansions satisfy a time-shift property. Furthermore, to guarantee the stability of the proposed filter, a sufficient condition for the BFLANN is also provided. The simulation results have shown that the performance of the nonlinear ANC system based on the BFLANN filter is better than that of the GFLANN and FLANN filters.

Funding

This work was partially supported by National Science foundation of P. R. China (Grant: 61671392) and by the Fund for the Basic Research Program of Sichuan province, China (Grant: 2013JY0036).

References

[1] Kuo SM, Morgan DR. Active noise control systems: algorithms and DSP implementations. New York: Wiley; 1996.
[2] Nelson PA, Elliot SJ. Active control of sound. New York: Academic; 1992.
[3] Elliott SJ, Nelson PA. Active noise control. IEEE Signal Process 1993;10(4):12–35.
[4] Kuo SM, Morgan DR. Active noise control: a tutorial review. Proc. IEEE 1999;87(6):943–73.
[5] Strauch P, Mulgrew B. Active control of nonlinear noise processes in a linear duct. IEEE Trans. Signal Process 1998;46(9):2404–12.
[6] Costa MH, Bermudez JCM, Bershad NJ. Stochastic analysis of the filtered-X LMS algorithm in systems with nonlinear secondary paths. IEEE Trans. Signal Process 2002;50(6):1327–42.
[7] Panda G, Das DP. Functional link artificial neural network for active control of nonlinear noise processes. International workshop on acoustic echo and noise control, Kyoto, Japan 2003:163–8.
[8] Snyder SD, Tanaka N. Active control of vibration using a neural network. IEEE Trans. NeuralNetw 1995;6(4):819–29.
[9] Chang CY. Neural filtered-U algorithm for the application of active noise control system with correction terms momentum. Digital Signal Process 2010;20(4):1019–26.
[10] Bouchard M, Pailard B, Dinh CTL. Improved training of neural networks for nonlinear active control of sound and vibration. IEEE Trans. Neural Networks 1999;10(2):391–401.
[11] Zhang QZ, Gan WS, Zhou YL. Adaptive recurrent fuzzy neural networks for active noise control. J. Sound Vib 2006;296(4):935–48.
[12] Mathews VJ, Sicuranza GL. Polynomial signal processing. New York: Wiley; 2000.
[13] Tan L, Jiang J. Adaptive volterra filter for active control of nonlinear noise processes. IEEE Trans. Signal Process 2001;49(8):1667–76.
[14] Das DP, Panda G. Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm. IEEE Trans. Speech Audio Process 2004;12(3):313–22.
[15] Das DP, Mohapatra SR, Routray A, Basu TK. Filtered-s LMS algorithm for multichannel active control of nonlinear noise processes. IEEE Trans. Speech Audio Process 2006;14(5):1875–80.
[16] Behera SK, Das DP, Subudhi B. Adaptive nonlinear active noise control algorithm for active headrest with moving error microphones. Appl Acoust 2017;123:9–19.
[17] George NV, Panda G. A robust filtered-s LMS algorithm for nonlinear active noise control. Appl Acoust 2012;73:836–41.
[18] George NV, Panda G. A particle-swarm-optimization-based decentralized nonlinear active noise control system. IEEE Trans Instrum Meas 2012;61(12):3378–86.
[19] Das DP, Moreau DJ, Cazzolato BS. Nonlinear active noise control for headrest using virtual microphone control. Control Eng Pract 2013;21(4):544–55.
[20] Das DP, Moreau DJ, Cazzolato BS. A nonlinear active noise control algorithm for virtual microphones controlling chaotic noise. J Acoust Soc Am 2012;132(2):779–88.
[21] Sicuranza GL, Carini A. On the BIBO stability condition of adaptive recursive FLANN filters with application to nonlinear active noise control. IEEE Trans Audio Speech Lang Process 2012;20(1):234–45.
[22] Sicuranza GL, Carini A. Adaptive recursive FLANN filters for nonlinear active noise control. IEEE International conference on acoustics 2011;125(3):4312–5.
[23] Zhao HQ, Zeng XP, Zhang JS. Adaptive reduced feedback FLNN filter for active noise control of nonlinear noise processes. Signal Process 2010;90(3):834–47.
[24] George NV, Panda G. On the development of adaptive hybrid active noise control system for effective mitigation of nonlinear noise. Signal Process 2012;92(2):509–16.
[25] George NV, Gonzalez A. Convex combination of nonlinear adaptive filters for active noise control. Appl Acoust 2014;76:157–61.
[26] George NV, Panda G. Active control of nonlinear noise processes using cascaded adaptive nonlinear filter. Appl Acoust 2014;74:217–22.
[27] Behera SK, Das DP, Subudhi B. Functional link artificial neural network applied to active noise control of a mixture of tonal and chaotic noise. Appl Soft Comput 2014;23:51–60.
[28] Sicuranza GL, Carini A. A generalized FLANN filter for nonlinear active noise control. IEEE Trans Audio Speech Lang Process 2011;19(8):2412–7.
[29] Kuo SM, Wu HT. Nonlinear adaptive bilinear filters for active noise control systems. IEEE Trans Circuits Syst-I: Regular Papers 2005;52(3):617–24.
[30] Tan L, Jiang J. Nonlinear active noise control using diagonal-channel LMS and RLS bilinear filters. In: IEEE 57th international midwest symposium on circuit & systems, College Station, Texas; August 2014. p. 789–92.
[31] Tan L, Jiang J, Wang L. Adaptive diagonal-channel bilinear filters for nonlinear active noise control. Int J Control Sci Eng 2014;4(2):27–35.
[32] Tan L, Dong C, Du S. On implementation of adaptive bilinear filters for nonlinear active noise control. Appl Acoust 2016;106:122–8.
[33] Zhao H, Zeng X, He Z, Li T, Jin W. Nonlinear adaptive filter-based simplified bilinear model for multichannel active control of nonlinear noise processes. Appl Acoust 2013;74(12):1414–21.
[34] Zhou D, Brunner V. Efficient adaptive nonlinear filters for nonlinear active noise control. IEEE Trans Circuits Syst-I 2007;54(3):669–81.
[35] Feintuch PL. An adaptive recursive LMS filter. Proc IEEE 1976;64(11):1622–4.
[36] Lee Junghsi, Mathews VJ. A Stability condition for certain bilinear systems. IEEE Trans Signal Process 1994;42(7):1871–3.