# Generalized Adaptive Exponential Functional Link Network Approximates

Vinu Sankar Sadasivan, Sankha Subhra Bhattacharjee, Vinal Patel, and Nithin V. George

***Abstract*—Algorithms designed under linear assumption are not efficient in modeling nonlinear systems. The adaptive exponential functional link network (AEFLN) was recently introduced to improve the modeling accuracy of nonlinear systems. The AEFLN, however, does include cross terms, which limits its modeling accuracy. To improve the modeling accuracy of nonlinear systems, a generalized functional expansion structure for AEFLN (GAEFLN) is developed in this paper. In order to reduce the computational complexity, a recursive generalized AEFLN (RGAEFLN) is also proposed. In addition, a nonlinear active noise control approach based on GAEFLN is also developed. Extensive simulation studies demonstrated our nonlinear system's superior modeling and active noise control capability.**

***Index Terms*—Nonlinear filter, active noise control, functional link network, linear-in-the-parameter nonlinear filter**

## I. Introduction

SEVERAL nonlinear architectures has been proposed in literature for the modelling of nonlinear systems [1], [2], [3], [4]. Nonlinear system modelling using multi layer perceptron (MLP) was proposed in [1], [3]. In the recent years, several single layer neural networks which can provide good modelling accuracy have been developed for modelling nonlinear systems, which also has much lower computational complexity compared to MLP. Linear-in-the-parameter (LIP) nonlinear filters are one such class of single layer nonlinear filters which has become quite popular. One of the widely studied LIP nonlinear filters are the functional link networks (FLNs). In a FLN, the input signal is expanded to a higher dimension where it becomes linear followed by traditional adaptive filters [5], [6], [7]. A FLN where Chebyshev polynomials were used for functional expansion of the input was proposed in [8]. Chebyshev FLN was seen to provide similar performance compared to MLP but with much less computational complexity. A trigonometric FLN (TFLN) which uses an expansion where the input signal is expanded via sinusoidal basis functions, was proposed in [7] for nonlinear active noise control (NANC). A FLN which uses Legendre polynomials for functional expansion of the input was proposed in [9] for the problem of nonlinear communication channel equalization, where Legendre FLN was seen to provide similar performance as that achieved by Chebyshev FLN. A generalized FLN (GFLN) was proposed in [10] for NANC, where in addition to the trigonometric terms, cross terms obtained from the input signal vector elements are also used in the functional expansion. Introduction of cross terms was seen to improve the convergence performance of GFLN compared to FLN.

Pure sinusoidal functional expansion in a TFLN cannot effectively model fast amplitude variations in natural sounds, like speech signals and better modelling accuracy can be achieved when exponentially varying sinusoids are used. With this motivation, the authors in [11] proposed an adaptive exponential FLN (AEFLN) that uses exponentially varying sinusoids with adaptive decay rate, for the functional expansion. AEFLN has been used in a range of scenarios such as, nonlinear system identification [11], NANC [11], [12], and nonlinear acoustic echo cancellation (NAEC) [13]. In [13], the authors proposed to use a recursive adaptive filter, where an infinite impulse response (IIR) filter follows the functional expansion and was reported to provide better convergence performance compared to AEFLN at a lower computational complexity. Traditional AEFLN as proposed in [11] does not include input cross terms in the functional expansion. In this paper, we propose a variable step size filtered-x recursive generalized adaptive exponential FLN (RGAEFLN) based nonlinear active noise canceller, which along with exponentially varying sinusoids, also includes input cross terms in the functional expansion followed by an adaptive IIR filter. The convergence performance of RGAEFLN is evaluated in comparison with GFLN and AEFLN for multiple nonlinear system identification and NANC scenarios with different system nonlinearities.

The rest of the paper is organized as follows. The proposed approach is described and the corresponding update rules are derived in Section II. In Section IV, a detailed simulation study is carried out and the results are discussed. The conclusions are presented in Section V.

## II. Proposed Method

In a traditional TFLN, the input signal $x(n)$ is expanded as in [7] where the expanded input vector $g(n)$ is of length $(2P+1)N+1$ and $P$ is the expansion order. For GFLN, the expanded input vector is given as in [10] where the expanded input vector $g(n)$ is of length $N^2+2N+1$. For AEFLN with a trainable adaptive exponential parameter $a(n)$, the expanded input vector is given as in [11] where the expanded input vector $g(n)$ is of length $(2P+1)N+1$. In recursive FLNs, the output signal produced by the the FLN model is fed back into the model along with the input helping the model in a faster convergence.

This work is supported by the Department of Science and Technology, Government of India under the Core Grant Scheme (CRG/2018/002919) and TEOCO Chair of Indian Institute of Technology Gandhinagar.

V. S. Sadasivan, S. S. Bhattacharjee, and N. V. George are with the Indian Institute of Technology Gandhinagar, Gujarat-382355, India (e-mail: {vinu.sankar, sankha.bhattacharjee, nithin}@iitgn.ac.in).

V. Patel is with the Indian Institute of Information Technology and Management, Gwalior-474015, India (email: vp@iiitm.ac.in)

The functional expansion for RGAEFLN is given by

$$\begin{aligned} g(n) = \; & \{1,\ x(n),\ x(n-1),\ \ldots,\ x(n-N+1), \\ & e^{-a(n)|x(n)|}\sin[\pi x(n)],\ e^{-a(n)|x(n-1)|}\sin[\pi x(n-1)], \\ & \ldots,\ e^{-a(n-N+1)|x(n)|}\sin[\pi x(n-N+1)], \\ & e^{-a(n)|x(n)|}\cos[\pi x(n)],\ e^{-a(n)|x(n-1)|}\cos[\pi x(n-1)], \\ & \ldots,\ e^{-a(n)|x(n-N+1)|}\cos[\pi x(n-N+1)], \\ & e^{-a(n)|x(n-1)|}x(n-1)\sin[\pi x(n)], \\ & e^{-a(n)|x(n-2)|}x(n-2)\sin[\pi x(n-1)],\ \ldots, \\ & e^{-a(n)|x(n-N+1)|}x(n-N+1)\sin[\pi x(n-N+2)], \\ & e^{-a(n)|x(n-1)|}x(n-1)\cos[\pi x(n)],\ \ldots, \\ & e^{-a(n)|x(n-N+1)|}x(n-N+1)\cos[\pi x(n-N+2)], \\ & y(n-1),\ y(n-2),\ \ldots,\ y(n-M),\ \ldots, \\ & e^{-a(n)|y(n-M+1)|}y(n-M+1)\sin[\pi y(n-M+2)], \\ & e^{-a(n)|y(n-M+1)|}y(n-M+1)\cos[\pi y(n-M+2)]\}^{\mathrm{T}} \end{aligned} \tag{1}$$

where $N$ and $M$ denote the lengths of the input and the output signals, respectively. The delayed signals $\{x(n),\ \ldots,\ x(n-N+1)\}$ and $\{y(n),\ \ldots,\ y(n-M+1)\}$ are used to generate the expanded layer. The length of the expanded signal $g(n)$ is found to be $L = N^2 + M^2 + 2(N+M) + 1$. The weight vector $w_g(n) = \{w_g^1(n),\ w_g^2(n),\ \ldots,\ w_g^L\}^{\mathrm{T}}$ is applied to get the output from the adaptive filter, expressed as $y(n) = g(n)^{\mathrm{T}} w_g(n)$.

## III. Adaptation rule

### A. Variable step size algorithm

To robustly compare various filters, we use a VSS algorithm proposed in [14]. This lets the model use an adaptive step size which increases or decreases according to the error of the system. The VSS algorithm is given by $\mu(n+1) = \alpha\mu(n) + \gamma e(n)^2$, where $\mu(n)$ is the step size of the adaptation rule and $e(n)$ is the error of the system at time step $n$. In all the simulations, we set $\alpha = 0.97$, $\gamma = 0.8 \times 10^{-4}$, and $\mu(1) = 0.01$ for all the models.

### B. System identification

In this section, we will discuss the adaptation rule used for updating the trainable filter parameters for system identification tasks. Figure 1 using LMS algorithm. For this system identification setup, let $d(n)$ be the output of the system which is to be modelled with RGAEFLN. The error of the system is given by $e(n) = d(n) - y(n)$. The adaptation rule for the weight vector is given by

$$\begin{aligned} w_g(n+1) &= w_g(n) - \mu_w(n)\Delta_w(n)/2 \\ &= w_g(n) + \mu_w(n)e(n)g(n) \end{aligned} \tag{2}$$

where $\Delta_w(n)$ is an estimate of the gradient of $E[e^2(n)]$ with respect to weight $w_g(n)$ and $\mu_w$ is the learning rate ($E[.]$ denotes the expectation operator). Similarly, for the adaptive exponential FLNs, the rule for updating the adaptive exponential terms $a(n)$ is given as

$$\begin{aligned} a(n+1) &= a(n) - \mu_a(n)\Delta_a(n)/2 \\ &= a(n) + \mu_a e(n)\frac{\partial g(n)^{\mathrm{T}}}{\partial a(n)} w_g(n) \end{aligned} \tag{3}$$

where $\Delta_a(n)$ is an estimate of the gradient of $E[e^2(n)]$ with respect to the weight $w_g(n)$ and $\mu_a$ is the learning rate.

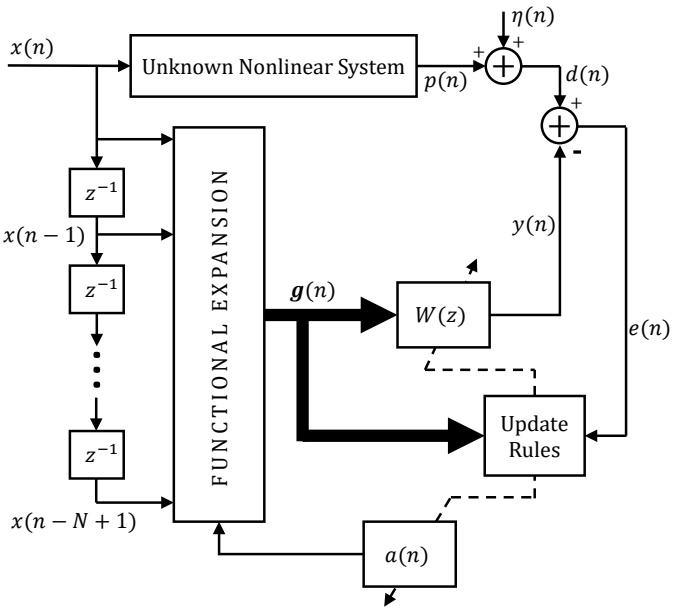


Fig. 1. Block diagram of AEFLN class of filters for nonlinear system identification.

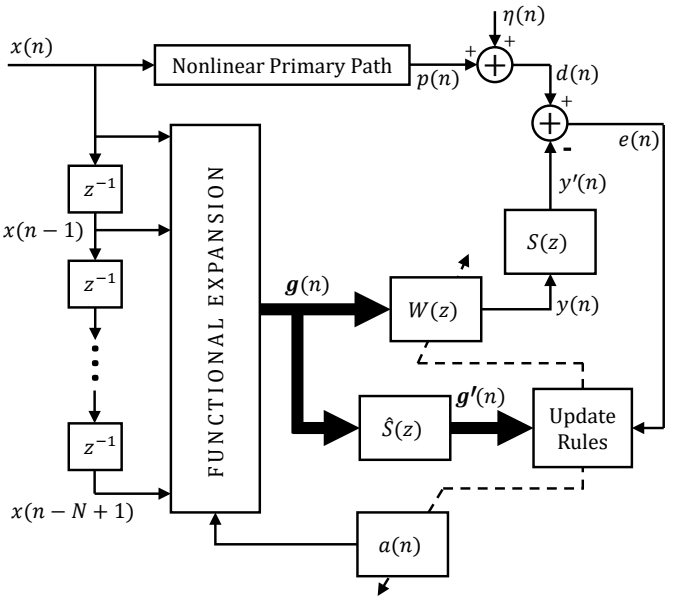


Fig. 2. Block diagram of filtered-x AEFLN class of nonlinear active noise cancellers.

### C. Active noise control

In this section, we will discuss the adaptation rule used for updating the trainable model parameters for ANC tasks.

Figure 2 shows the high-level block diagram of a ANC system that updates its weights using FxLMS algorithm. For this ANC setup, $P(z)$ is the transfer function of the primary path and $S(z)$ is the transfer function of the secondary path ($s(n)$ is the impulse response of the secondary path). $x(n)$ and $y(n)$ are the reference and controller output signals, respectively. $W(z)$ is the transfer function of the adaptive controller and $\hat{y}(n)$ is the secondary output. $d(n)$ is the reference signal sensed at the error microphone after it is affected by the primary path. The error noise measured by the error microphone is given as $e(n) = d(n) - \hat{d}(n)$, where $\hat{d}(n) = y(n) * s(n)$ and $*$ denotes the linear convolution operator. The adaptation rule for the weight vector is similar to that in equation 2 except that $\Delta_w(n) = 2e(n)[g(n) * s(n)]$

$$w_g(n+1) = w_g(n) + \mu_w(n)e(n)[g(n) * s(n)]. \quad (4)$$

Similarly, for adaptive exponential FLNs, the rule for updating the adaptive exponential term $a(n)$ is given similar to that in equation 3 except that $\Delta_a(n) = 2e(n)[s(n) * \frac{\partial g(n)}{\partial a(n)}]^{\mathrm{T}} w_g(n)$,

$$a(n+1) = a(n) + \mu_a(n)e(n)[s(n) * \frac{\partial g(n)}{\partial a(n)}]^{\mathrm{T}} w_g(n). \quad (5)$$

## IV. Simulation Study

In this section, we extensively compare the performances of GFLN, AEFLN, and AEGFLN filters for two different application areas. Examples 1 and 2 compare the models for nonlinear system identification tasks while examples 3, 4, and 5 compare models in NANC scenarios. All the three filters use the same order of memory for weight. For GFLN and AEFLN, $N = 10$ for all the examples. For AEFLN, $P$ takes the value 6. $N$ and $M$ are set to be 9 and 4 for RGAEFLN. The reference signal $x(n)$ is random noise generated with a uniform distribution between $-0.5$ and $+0.5$. White Gaussian noise of 30 $dB$ is added to the reference signal, after its interaction with the primary path. The initial values of all adaptive exponential terms are set to 1. All the simulations are run for 25 independent trials. We use average mean square error of last 1000 iterations (MSE) to compare the performance of the models. It is defined as MSE $= 10\ \log_{10}\{E[e^2(n)]\}$.

### A. Example 1

In this example, we have try to model a system that changes after every $6 \times 10^6$ samples similar to [11]. A soft clipping nonlinear system which represents the saturation effect of a loudspeaker [15], [16] is modelled in Stage A. The input output relation for Stage A is given by

$$g_1(n) = \beta.\Big\{\frac{1}{1+e^{-\rho q(n)}} - \frac{1}{2}\Big\} \quad (6)$$

where $\beta$ is the gain of the system. For this simulation, we set $\beta = 2$. $\rho$ represents slope of a sigmoid function is given by

$$\rho = \begin{cases} 4, & q(n) > 0 \\ 0.5, & q(n) \leq 0 \end{cases} \quad (7)$$

where $q(n) = 1.5x(n) - 0.3x^2(n)$. The input output relation for Stage B which simulates the nonlinear nature of a loudspeaker is given by

$$g_2(n) = \begin{cases} \frac{1}{18\zeta}x(n), & 0 \leq \mid x(n) \mid < \zeta \\ \mathrm{sign}[x(n)]\frac{3-[2-|x(n)|/\zeta]^2}{36}, & \zeta \leq \mid x(n) \mid < 2\zeta \\ \frac{\mathrm{sign}[x(n)]}{12}, & 2\zeta \leq \mid x(n) \mid < 0.5 \end{cases} \quad (8)$$

where $0 < \zeta \leq 0.5$ [17]. In this experiment, we set $\zeta = 0.1$.

The input output relation for Stage C is given by

$$\begin{aligned} g_3(n) = \ & \exp[0.5x(n)]\{\sin[\pi x(n)] + 0.3\ \sin[3\pi x(n)] \\ & + 0.1\ \sin[0.5\pi x(n)]\}. \end{aligned} \quad (9)$$

The learning curves are shown in Figure 3. The MSE values obtained for GFLN, AEFLN, and RGAEFLN, respectively, are; $-19.9032\ dB$, $-20.0619\ dB$, and $-19.9645\ dB$ (Stage A); $-5.742\ dB$, $-5.3882\ dB$, and $-5.9528\ dB$ (Stage B); $-16.5913\ dB$, $-16.4599\ dB$, and $-20.002\ dB$ (Stage C).

### B. Example 2

The input output relation for this example is given by

$$\begin{aligned} g(n) = \ & 0.6\sin^3[\pi x(n)] - \frac{2}{x^3(n)+2} \\ & - 0.1\cos[4\pi x(n-4)] + 1.125. \end{aligned} \quad (10)$$

The learning curves are shown in Figure 4. The MSE values obtained for GFLN, AEFLN, and RGAEFLN are $-23.0483\ dB$, $-25.7925\ dB$, and $-27.8785\ dB$, respectively.

### C. Example 3

In this example, Volterra models are used to describe primary and secondary paths as

$$\begin{aligned} d(n) = \ & x(n) + 0.8x(n-1) - 0.3x(n-2) + 0.4x(n-3) \\ & - 0.8x(n)x(n-1) - 0.9x(n)x(n-2) \\ & + 0.7x(n)x(n-3) \\ \hat{d}(n) = \ & y(n) + 0.35y(n-1) - 0.09y(n-2) \\ & - 0.5y(n)y(n-1) + 0.4y(n)y(n-2) \end{aligned} \quad (11)$$

The learning curves for the models are shown in Figure 5. The MSE values obtained for GFLN, AEFLN, and RGAEFLN are $-23.3678\ dB$, $-17.9749\ dB$, and $-24.2573\ dB$ respectively.

### D. Example 4

Now we consider primary path

$$\begin{aligned} d(n) = \ & x(n) + 0.8x(n-1) - 0.3x(n-2) + 0.4x(n-3) \\ & - 0.8x(n)x(n-1) - 0.9x(n)x(n-2) \\ & + 0.7x(n)x(n-3) - 3.9x^2(n-1)x(n-2) \\ & - 2.6x^2(n-1)x(n-3) - 2.6x^2(n-2)x(n-3) \end{aligned} \quad (12)$$

and secondary path same as in Example 1. The learning curves for the models are shown in Figure 6. The MSE values obtained for GFLN, AEFLN, and RGAEFLN are $-19.6075\ dB$, $-15.3984\ dB$, and $-21.0366\ dB$ respectively.

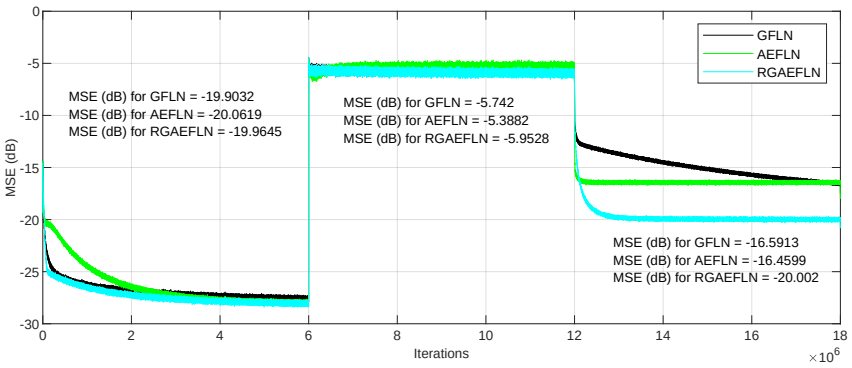


Fig. 3. Example 1: Comparing convergence characteristics of GFLN, AEFLN, and RGAEFLN.

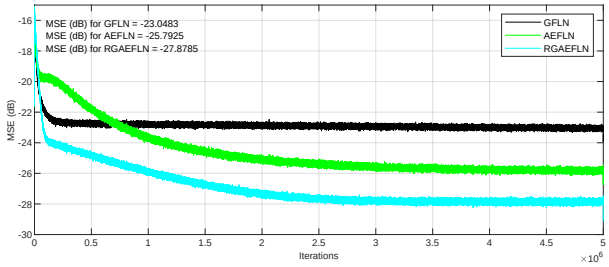


Fig. 4. Example 2: Comparing convergence characteristics of GFLN, AEFLN, and RGAEFLN.

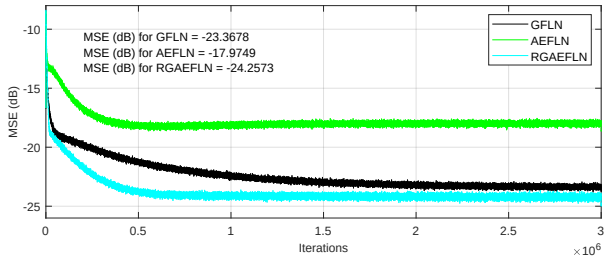


Fig. 5. Example 3: Comparing convergence characteristics of GFLN, AEFLN, and RGAEFLN.

### *E. Example 5*

We refer here to the same primary path in Example 1. The secondary path is modelled using a memoryless nonlinear Hammerstein filter given as $w(n) = \tanh[y(n)]$, followed by the linear filter

$$\hat{d}(n) = w(n) - 0.2w(n-1) + 0.05w(n-2) \qquad (13)$$

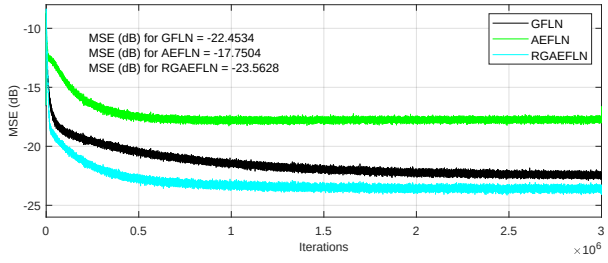


Fig. 6. Example 4: Comparing convergence characteristics of GFLN, AEFLN, and RGAEFLN.

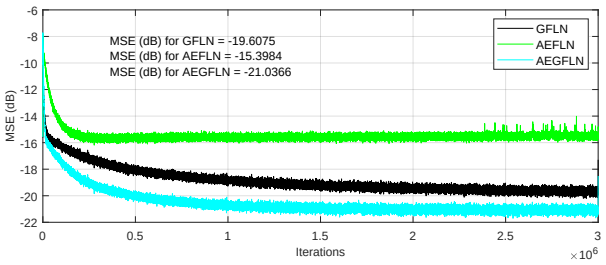


Fig. 7. Example 5: Comparing convergence characteristics of GFLN, AEFLN, and RGAEFLN.

These paths can be used to describe the nonlinearities in loudspeaker and power amplifier at the controller output [10]. The learning curves for the models are shown in Figure 7. The MSE values obtained for GFLN, AEFLN, and RGAEFLN are $-22.4534\ dB$, $-17.7504\ dB$, and $-23.5628\ dB$ respectively.

## V. Conclusion

## References

[1] Narendra, K. S. and Parthasarathy, K. Identification and control of dynamical systems using neural networks, *IEEE Transactions on Neural Networks*, **1** (1), 4–27, (1990).

[2] Scarpiniti, M., Comminiello, D., Parisi, R. and Uncini, A. Nonlinear spline adaptive filtering, *Signal Processing*, **93** (4), 772–783, (2013).

[3] Lu, S. and Basar, T. Robust nonlinear system identification using neural-network models, *IEEE Transactions on Neural Networks*, **9** (3), 407–429, (1998).

[4] Peng, L. and Ma, H. Design and implementation of software-defined radio receiver based on blind nonlinear system identification and compensation, *IEEE Transactions on Circuits and Systems I: Regular Papers*, **58** (11), 2776–2789, (2011).

[5] Pao, Y., *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley, Reading, MA (1989).

[6] Patra, J. C., Meher, P. K. and Chakraborty, G. Nonlinear channel equalization for wireless communication systems using Legendre neural networks, *Signal Processing*, **89** (11), 2251–2262, (2009).

[7] Das, D. P. and Panda, G. Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm, *IEEE Transactions on Speech and Audio Processing*, **12** (3), 313–322, (2004).

[8] Patra, J. C. and Kot, A. C. Nonlinear dynamic system identification using chebyshev functional link artificial neural networks, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **32** (4), 505–511, (2002).

[9] Patra, J. C., Chin, W. C., Meher, P. K. and Chakraborty, G. Legendre-flann-based nonlinear channel equalization in wireless communication system, *2008 IEEE International Conference on Systems, Man and Cybernetics*, pp. 1826–1831, IEEE, (2008).

[10] Sicuranza, G. L. and Carini, A. A generalized flann filter for nonlinear active noise control, *IEEE Transactions on Audio, Speech, and Language Processing*, **19** (8), 2412–2417, (2011).
[11] Patel, V., Gandhi, V., Heda, S. and George, N. V. Design of adaptive exponential functional link network-based nonlinear filters, *IEEE Transactions on Circuits and Systems I: Regular Papers*, **63** (9), 1434–1442, (2016).
[12] Patel, V., Pradhan, S. and George, N. V. Collaborative adaptive exponential linear-in-the-parameters nonlinear filters, *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 2694–2698, IEEE, (2017).
[13] Zhang, S. and Zheng, W. X. Recursive adaptive sparse exponential functional link neural network for nonlinear aec in impulsive noise environment, *IEEE transactions on neural networks and learning systems*, **29** (9), 4314–4323, (2017).
[14] Kwong, R. H. and Johnston, E. W. A variable step size lms algorithm, *IEEE Transactions on Signal Processing*, **40** (7), 1633–1642, (1992).
[15] Scarpiniti, M., Comminiello, D., Parisi, R. and Uncini, A. Novel cascade spline architectures for the identification of nonlinear systems, *IEEE Transactions on Circuits and Systems I: Regular Papers,*, **62** (7), 1825–1835, (2015).
[16] Comminiello, D., Scarpiniti, M., Azpicueta-Ruiz, L. A., Arenas-García, J. and Uncini, A. Functional link adaptive filters for nonlinear acoustic echo cancellation, *IEEE Transactions on Audio, Speech, and Language Processing*, **21** (7), 1502–1512, (2013).
[17] Comminiello, D., Scarpiniti, M., Azpicueta-Ruiz, L. A., Arenas-García, J. and Uncini, A. Nonlinear acoustic echo cancellation based on sparse functional link representations, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **22** (7), 1172–1183, (2014).