ADAPTIVE NOISE CANCELLATION USING DYNAMIC FUZZY NEURAL NETWORKS ALGORITHM

Meng Joo Er Member, IEEE and Aung Min Sia

Intelligent Machine Research Laboratory
School of Electrical and Electronic Engineering
Nanyang Technological University
Block S1, Nanyang Avenue, Singapore 639798
Email: emjer@ntu.edu.sg

Abstract: In this paper, Adaptive Noise Cancellation using Dynamic Fuzzy Neural Networks (D-FNN) algorithm is attempted. The D-FNN algorithm is a hierarchical on-line self-learning algorithm based on Extended Radial Basis Function Neural Networks. The salient features of the algorithm are: (1) It provides an efficient learning method; (2) The weights are modified by the Recursive Least Square method; (3) No iteration is needed. Simulation studies and comparisons with other algorithms show that the D-FNN algorithm is superior in terms of simplicity of structure, learning efficiency and performance in canceling noise adaptively. Copyright@2002 IFAC

Keywords: Dynamic fuzzy neural networks; adaptive noise cancellation.

1. INTRODUCTION

In the area of signal processing, extracting the true information signal from its corrupted signal is one of the challenging problems. One of the common methods is to pass the noisy signal through a filter which is used to suppress the noise and to recover the original signal. There are many approaches to achieve the objective of noise cancellation or noise suppression. In the context of filters, frequency-selective filters such as band-pass filters and band-limit filters are widely used. For both cases, a priori knowledge of both the signal and the noise are important. However, in the scenario where the noise overlaps with the signal, band-pass or band-limited filters can no longer be used.

Adaptive filters are designed in order to adjust the system parameters automatically so that no a priori knowledge of signal and noise are required. In Mastorocostas and Theocharis (2000), adaptive noise cancellation was studied using the Recurrent-Neural-Network-Based Dynamic Fuzzy Neural Network. It consists of three different parts, namely the premise part, the consequent part and the defuzzification part. The premise part contains three layers: the input layer, the fuzzification layer and the rule layer. The consequent part contains \( r \) sub-models. Each sub-model is a recurrent neural network in the form of \( m-H_n-1 \), where \( r \) is the number of sub-models, \( m \) is the number of neurons in the input layer and \( H_n \) is the number of neurons in the hidden layer. The hidden and output layers consist of Generalized Frasconi-Gori-Soda (G-FGS) neurons. According to Mastorocostas and Theocharis (2000), the G-FGS neurons exhibit improved identification characteristics.

In Piche (1994), the noise filter was implemented by a three-layer feed-forward neural network. In this approach, the network structure was kept fixed and the parameter were updated using the Steepest Descent Algorithm. The approach employed 20 neurons and was required to train about 6000 iterations.

In Lin and Juang (1997), the Adaptive Neural Fuzzy Filter (ANFF) was developed. In this approach, the Fuzzy Adaptive Resonance Theory (ART) is applied to perform fuzzy clustering in the input-output spaces and obtain appropriate fuzzy logic rules dynamically. The authors have shown that the performance of
adaptive fuzzy filters is better than linear filters of different orders.

In this paper, Adaptive Noise Cancellation using the D-FNN algorithm developed in Er and Wu (2001) is attempted. The main objective of this paper is to demonstrate that the D-FNN algorithm can be employed for adaptive noise cancellation. Simulation studies and comparisons with other approaches show that our approach is superior.

The paper is organized into five sections. In Section II, the problem of adaptive noise cancellation is formulated. Section III presents two simulation studies to demonstrate excellent performance of the proposed approach. In Section IV, a comparative study of our result with other results is made. Section V concludes the paper.

2. ADAPTIVE NOISE CANCELLATION PROBLEM

The typical structure of noise cancellation system is depicted in Fig. 2.1.

\[ d(k) = s(k) + n(k) \]  

From (2.1), we have

\[ e(k) = d(k) - \hat{y}(k) = s(k) + n(k) - \hat{y}(k) \]  

From Eq.(2.3), we can see that if the filter or the model output \( \hat{y}(k) \) can replicate the noise \( n(k) \) i.e. \( \hat{y}(k) \rightarrow n(k) \), then the error \( e(k) \) corresponds to the information signal. This problem can also be seen as a system identification problem. In this case, the channel dynamics \( T(\cdot) \) is to be identified. The system identification problem is depicted in Fig.2.2.

\[ e^2(k) = (d(k) - \hat{y}(k))^2 = (s(k) + n(k) - \hat{y}(k))^2 \]  

From Eq.(2.2), we have

\[ E\{e^2(k)\} = E\{(s(k) + n(k) - \hat{y}(k))^2\} \]  

which in turn gives

\[ E\{s^2(k)\} + E\{(n(k) - \hat{y}(k))^2\} + 2E\{s(k)n(k)\} - 2E\{s(k)\hat{y}(k)\} \]

In Fig. 2.1, the desired information signal \( s(k) \) is corrupted by the noise \( n(k) \). \( T(\cdot) \) represents the nonlinear characteristic of the passage that the noise passes through. Therefore, the measured signal \( d(k) \) is:

\[ d(k) = s(k) + n(k) \]  

Here, the information signal and noise signal are assumed to be uncorrelated. Furthermore, it is assumed that the channel characteristic \( T^{-1} \) can be estimated. The case where the reference noise \( x(k) \) will pass through the nonlinear channel \( T(\cdot) \) is considered here. In this problem, both the measured signal \( d(k) \) and the reference signal are assumed to be available.
Since the information signal is not correlated with the noise $n(k)$ or its estimate $\hat{y}(k)$, the last two terms in Eq.(2.5) are equal to zero. Eq.(2.5) can be rewritten as

$$E\{e^2(k)\} = E\{s^2(k)\} + E\{(n(k) - \hat{y}(k))^2\}$$

(2.6)

The objective is to minimize $E\{(n(k) - \hat{y}(k))^2\}$ and this leads to minimizing $E\{e^2(k)\}$. It is also not difficult to see that when $E\{(n(k) - \hat{y}(k))^2\} \rightarrow 0$, $e(k)$ is the exact recovered signal of information $s(k)$. In this paper, two examples on noise cancellation which have been studied in (Mastorocostas and Theocaris, 2000, Piche, 1994 and Lin and Juang, 1997) are used in order to evaluate the performance of the D-FNN algorithm.

3. SIMULATION STUDIES

3.1 Example 1

In this example, the noise filter is the nonlinear IIR filter. The D-FNN is trained with 1000 data set to be used as a noise canceller. The original information signal is:

$$s_k = 0.25\cos(0.4k)$$

(3.1)

The reference signal $x(k)$ is a uniformly distributed random variable with the range of –1.0 to 1.0. This noise passes through the nonlinear channel and $n(k)$ is:

$$n(k) = x(k) + f_n(n(k-1))$$

(3.2)

where

$$f_n(n(k-1)) = 0.5\exp\left(-\frac{(n(k-1)-1.0)^2}{0.67}\right) - 0.5\exp\left(-\frac{(n(k-1)+1.0)^2}{0.67}\right)$$

(3.3)

It should be noted that Eq.(3.1) contains a nonlinear feedback term.

Fig. 3.1(a). Corrupted signal and recoverd signal for 100 test data.

The D-FNN algorithm is trained for 900 data points and Fig. 3.1(a) shows the corrupted signal and recovered signal for 100 test data points. The original information signal and the recovered signal are shown in Fig. 3.1(b). From Fig. 3.1(b), we can see that the D-FNN can reproduce the original signal very well. For this example, the D-FNN only needs two hidden neurons and Fig. 3.1(c) depicts the growth of neurons. This figure illustrates that the D-FNN algorithm can optimize the network parameters in a very short time which is less than 80 sample data in this case. Fig. 3.1(d) and Fig. 3.1(e) shows the model output error for the entire sample data set and the root mean squared error respectively.
Figure 3.1(b). Information signal and recovered signal

Figure 3.2(a). Information signal (music)

Figure 3.2(b). Noise signal (bird’s chirping)

Figure 3.2(c). Distorted signal

Figure 3.2(d). Original signal and recovered signal for 100 test data

\[ x(k) = T(n(k)) = \frac{8\sin(n(k)n(k-1)n(k-2))}{1+[n(k-1)]^2+[n(k-2)]^2} \]  

(3.4)

From above equation, we can see that the output exclusively depends on inputs. The information and noise signals are shown in Fig. 3.2(a) and 3.2(b) respectively. The distorted signal is depicted by Fig. 3.2(c). The final number of neurons is 6 and the optimal parameters are achieved when the training reaches about 2200 sample data. The original information signal and the recovered signal for 100 test data are shown in Fig. 3.2(d).

3.2 Example 2

In this example, both information and noise signals are real-world audio signals obtained from MATLAB® sound files handel.mat and chirp.mat which is a piece of music of The Hallelujah Chorus and a bird’s chirping respectively. The piece of music is selected as an information signal and the bird’s chirping is used as the noise. As in (Mastorocostas and Thecharis, 2000), the channel characteristic is selected to be static, i.e.
As mentioned in Er and Wu (2001) the adaptive capability of LLS method would decrease when more sample data are collected. That can be seen in this example. Here, although the output exclusively depends on the inputs, the inputs of the model are time varying. Although the D-FNN can identify the nonlinear characteristics of the channel well and adaptive noise cancellation is demonstrated, the delay of input feedback does affect the performance of the D-FNN. In order to alleviate this problem, alternative methods of learning are proposed. One approach is to use a sliding window which eliminates the effect of old training data. Another alternative approach is to use the RLS method (Kalman Filter) which is described by

\[ W_i = W_{i-1} + S_i \Psi_i^T (T_i - \Psi_i W_{i-1}) \]

\[ S_i = S_{i-1} - \frac{S_{i-1} \Psi_i^T \Psi_i S_{i-1}}{\lambda + \Psi_i S_{i-1} \Psi_i} \]  

(3.5)

where the initial conditions are \( W_0 = 0 \) and \( S_0 = \mu I \). \( S_i \) is the error covariance matrix for the \( i \)th observation, \( \Psi_i \) is the \( i \)th column of \( \Psi \), \( W_i \) is the coefficient matrix after the \( i \)th iteration, \( \mu \) is a positive large number and \( I \) is the identity matrix.

Simulation study was done using the RLS method with different forgetting factors. The results show that the Kalman Filter method can adapt the network structure well and compensates the effect of delayed input. Due to the coarse learning process during the first few hundreds sample data, the noise could not be cancelled effectively in both methods. The D-FNN is trained with 5000 training data for both methods and the results are summarised in Table 3.1.

<table>
<thead>
<tr>
<th>Example</th>
<th>No of Training data</th>
<th>Performance Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example 1</td>
<td>1000</td>
<td>2.0788 e –16</td>
</tr>
<tr>
<td>Example 2</td>
<td>5000</td>
<td>0.0392</td>
</tr>
<tr>
<td>Example 2 with Kalman Filter</td>
<td>5000</td>
<td>0.0313</td>
</tr>
</tbody>
</table>

Clearly, the smaller the value of PI is, the better performance it is. We can see that for Example 1 which has very small PI and the output of the model is exactly the replica of the desired one. However, for Example 2, the performance starts to decrease as more samples are taken. This is due to the effect of delayed inputs and its time-varying characteristic. From the simulations, it is evident that the PI can be improved using the Kalman filter approach. In Fig. 3.3(a), the recovered signal and the original signal
for the last 400 data points which were obtained by the D-FNN with the forgetting factor $\lambda = 0.6$ are shown.

4. DISCUSSIONS

In Example 1, a nonlinear filter with one delayed output feedback is simulated. In this problem, the nonlinear channel characteristic is simulated by squared functions of delayed outputs of the filter. Although it is highly nonlinear, the D-FNN algorithm can identify the system very well with very short training time. In (Piche, 1994) a fixed structure feedforward neural network was used to implement the nonlinear filter. It employed 20 neurons and the structure of the neural network was defined a priori. In our case, the D-FNN generates only two neurons and no a priori knowledge of the system is required to implement the adaptive filter.

In Example 2, both information and noise signals are real-world audio signals. The channel characteristic is represented by Eq. (3.4). Note that the characteristic of the channel is static since the output exclusively depends on the inputs. The D-FNN is trained for 5000 sample data and the optimal structure with 6 hidden neurons is obtained at the 2200th sample data.

It should be highlighted that the dynamics of the system are time dependent and outputs of the system depend on delayed inputs and outputs. From the simulation results, it is evident that there is a phase lead of a few sampling times which are introduced by the nonlinear channel dynamics.

Furthermore, the channel dynamics includes delayed inputs in the form of: $T(n(k-1),n(k-2),n(k-3))$. By shifting the recovered signal for 2 sample periods, the compensated signal is plotted together with the information signal in Fig. 3.4.

![Image](image_url)

Information signal ( )
Compensated recovered signal ( )

Fig. 3.4. Information signal and compensated recovered signal

5. CONCLUSIONS

In this paper, the D-FNN algorithm based on Extended RBF Neural Network is applied to cancel noise adaptively. We show that the D-FNN algorithm has better learning algorithm and less difficulties in terms of structure identification and parameters updating than the other methods. Therefore, the burden of computing is also relatively smaller than other methods. By using RBF function approach, there is no need to compute fuzzy membership function by trial and error. This leads to greater simplification in network learning and updating scheme. As foreshadowed earlier, the performance of the D-FNN decreases when more sample data are taken to identify the time-varying dynamics. In this paper, the improvement is made by applying the RLS method instead of the LLS method for weights updating. Other methods that can eliminate the effect of delayed inputs and outputs in channel dynamics should be considered in future research. In summary, we have demonstrated via two simulation studies that excellent performance can be achieved by applying the D-FNN algorithm in adaptive noise cancellation. This work serves as a basis for applying the D-FNN algorithm for real-time adaptive noise cancellation.

REFERENCES


