Adaptive Noise Cancellation Based on Multilayer Perception Neural Network Approach

Thamer M. Jamel AL-Anbaky*
Received on:11/2/2004
Accepted on:7/9/2004

Abstract

The Adaptive Noise Canceller (ANC) was achieved using many conventional techniques such as the adaptive FIR filler that uses Least Mean square (LMS) algorithm. In the past few decades, Neural Networks (NN) have been extensively applied to many signal processing problems due to its activity in forming complex decision regions. In this paper, suitable learning algorithms such as Back Propagation (BP) and Levenberg-Marquardt (LM) algorithms, which train the Multi-Layer Perceptron (MLP) neural network, were applied and tested for noise cancellation application. The simulation result shows fast convergence of LM algorithm over BP algorithm for different nodes size of NN system.

الخلاصة

تعمل المنظومة المتكيفة لالغاء الضوضاء بعدة تقنيات تقليدية مختلفة، منها تقنية المرشحات المتكيفة من نوع " FIR " والتي تستخدم خوارزمية " اقل معدل للخطأ " (LMS). في السنوات الاخيرة برزت تقنية استخدام الشبكات العصبية في العديد من تطبيقات معالجة الاشارة الرقمية، وذلك لفاعلية هذه الشبكات في تشكيل مناطق قرار معقدة، وبالتالي كفاءها العالية في الكثير من التطبيقات (التمييز ، والتصنيف وغيرها) .

في هذا البحث، تم استخدام هذه التقنية (اي الشبكات العصبية) في مجال عمل منظومة الغاء الضوضاء المتكيفة، إذ تم اعتماد شبكة " بيرسبترون " العصبية ذات الطبقات المتعددة ، كذلك تم تعليم هذه الشبكة عن طريق خوارزمية الـ " BP " وخوارزمية الـ " LM ". نتائج المحاكاة بالحاسوب لعمل المنظومة باستخدام هذه التقنية أثبتت كفاءة وفاعلية هذه التقنية في الغاء الضوضاء وكذلك السرعة العالية لخوارزمية الـ " LM " مقارنة مع خوارزمية الـ " BP " .

*University of Technology – Department of Electrical and Electronic engineering. Baghdad, Iraq.

1. Introduction

ANC system is regarded as one of the methods used for signal (speech) enhancement and an attempt to reduce the additive noise, which may arise from different sources. Noise canceling is an adaptive system that makes use of an auxiliary or reference input as shown in Fig (1) [1]. In this Figure, ANC has two inputs called primary and reference inputs. The reference input is filtered and subtracted from a primary input, which contains both signal (speech) and noise. As a result, the primary noise is attenuated or eliminated by cancellation. This filtering and subtraction must be controlled by an appropriate adaptive process in order to obtain noise reduction with little risk of distorting the signal or increasing the noise level. [2]. A wide variety of algorithms are used by ANC to adjust their impulse response. The LMS algorithm is regarded as a special case of the gradient search algorithm, which was developed by Widrow and Hoff in 1959 [3]. This algorithm is often used for the adaptation of the ANC system because it is easy to implement and require small number of calculations. An alternate new approach to the conventional adaptive system is Neural Network which is non-linear adaptive system that has the potential to push the technology barrier beyond conventional approaches [4]. N.N models have greatest potential in areas, such as adaptive signal (speech) processing, due to their massive parallelism, high computation rates and its provision a greater degree of fault tolerance [5]. One of this digital signal processing applications was a non-linear adaptive filter which use multi-layer feed forward neural networks as a filter structure instead of transversal or recursive filter. The Neural Networks can be considered as a class of non-linear adaptive filters, which can be trained by several Neural Networks algorithms. This paper proposes two Neural Networks algorithms, which are back propagation (BP) and Levenberg-Marquardt (LM) algorithms that are used to train the Multi-Layer Perceptron (MLP) Neural Network.

2. Adaptive noise cancellation based on MLP Neural Net

Multi-Layer Perceptrons are feed-forward nets (MLFF) with one or more layers of nodes between the input and output nodes. These additional layers contain hidden units or nodes. Fig (2) shows the structure of MLP Neural Net. Lippman [4] shows that no more than three layers are required to generate arbitrarily complex decision regions for one output when hard-limiting nonlinearities are used. These nets can be trained with the following algorithms:-

2.1 Back Propagation training algorithm

Back Propagation is the most widely training algorithm used in training the multilayer Feed Forward Neural Network (MLFF), which can attack any problem that requires pattern mapping. By giving an input pattern, the network produces an associated output pattern [5]. The conceptual basis of Back Propagation (BP) was first presented in 1974 by David Parker and presented to a wide readership in 1986 by Rumelhart and McClelland. Since its invention, BP algorithm has been enhanced by adding several choices of design, such as the number of layers and nodes in the hidden layer, variable learning rate, momentum term, and dynamic BP, with the addition of error approximation for non-linear systems. These enhancements undoubtedly have increased the efficiency and the convergence speed of the algorithms. BP algorithm is one of the easiest networks to understand its learning update procedure, and is intuitively appealing because it is based on a relatively simple concept. If the network gives the wrong answer, then the weight is corrected so that the error is lessened and a result future response of the network is more likely to be correct.

2.2 Levenberg-Marquardt Algorithm

Levenberg-Marquardt (LM) is a training algorithm used to train MLFF network, based on nonlinear optimization technique by minimizing the sum of squares of error (SSE). The Levenberg-Marquardt algorithm is often superior to other training algorithms in off-line applications. This motivated the proposal of using the algorithm to train the Neural Networks.

LM algorithm is an implementation of Levenberg (1944). "A method for the solution of certain problem in least squares method" and Marquardt (1963) "Least square estimation of nonlinear parameter ". [6]

LM algorithm has better convergence properties than the other two methods, i.e. steepest-descent and Gauss Newton algorithms, and it is well known that it is the best choice in off-line training of Neural Networks. The reason is that the algorithm disregards the directions in the parameter space, which influence the criterion marginally. One advantage of LM is that not all the iterations in LM algorithm are used to update the network's parameters, but instead, only the iterations, which decrease the error, are used in updating the network parameters. Therefore, the error will never increase through the learning of the process, and hence it will have a stair-like performance surface.

Simulation results showed that, LM training algorithm could reach any degree of accuracy with more degree of freedom in the hidden layers.

3. Simulation results

In this section, two-input ANC, shown in Fig (1), is used which trained with BP and LM algorithm shown in Tables (1&2) respectively with the following conditions: -

a) Desired signal is speech signal with sampling frequency of 8 kHz and is applied to the primary input of ANC system as shown in Fig (3). Noise path filler between primary and input is 2^{nd} order digital Butterworth Low Pass Filter (LPF) with cut-off frequency of 0.5 KHZ which represents a real model used in the Driver voice communication [7].
b) White Gaussian noise with variance one and zero means it was applied to the reference input of ANC as shown in Fig (4).

Figures (5, 6, and 7) show mean square Error (MSE) for the BP and LM algorithm with different number of nodes in the hidden layer. This variance in the number of nodes provides the selection of the approximation size of the network by observing the behavior of the network in terms of number of nodes and fast convergence. As shown in Figures (5, 6, and 7), the MSE curves will be initiated with total power summation of speech and noise signals. Then the noise power will be cancelled gradually until the system converges to the desired speech power signal, which is equal to 0.1 in this simulation. The total averaged cancelled noise power can be calculated from these figures and it is equal to approximately 20 db. In addition, these figures show that BP algorithm converges within 200, 100, and 75 iterations for the number of nodes 10, 30, and 40 respectively. While LM algorithm converges within 6, 5, and 4 iterations with corresponding number of nodes in the hidden layer. These results show the efficient and good performance of N.N in the ANC application as an alternative approach to the conventional techniques such as adaptive FIR filtering.

4. Conclusions

In this paper, two-input ANC system based on MLP Neural Net-based approach was proposed. Two training algorithms (BP and LM) were proposed to train non-linear adaptive filtering application as ANC system. LM algorithm shows faster convergence and lower level of miss adjustment in the steady state than the BP algorithm. These two algorithms succeed to cancel about 20 db of the additive noise power. In addition, the design of the Neural Network system depends on the behavior of the network to a specific data, i.e. the number of nodes in the hidden layer). Also choosing the number of nodes in the hidden layer is trade-off between the requirement of fast convergence and lower miss adjustment in the steady state. However, the proposed algorithms reduce the trade-off requirements.

5. References

1. B. Widrow and Samuel D. Stearns, "Adaptive signal processing" Prentic Hall, Inc., 1985.
2. S.F. Boll, and D.C. publisher, "Suppression of acoustic noise in speech using two microphones ANC ", ICEE Trans. On ASSP-28, No.6, pp.752-753, Dec. 1980.
3. M.S. Alunes, "Comparison of noise speech enhancement algorithms in terms of LPC- perturbation", ICEE Trans. On ASSP-37, No.1, pp.121-125, Jan. 1989.
4. Don R., Hush and Bill G. Horne, "Progress in Supervised Neural Networks", ICEE signal magazine, pp.8-39, Jan. 1993.
5. Judith E. Danyhoff, "Neural Network architectures ", Von nostrana Reinhold, New York 1995.
6. Laster S.H. Ngia and Jonas Sj Ö berg, "Efficient training of Neural Nets for non-linear adaptive filtering using a recursive levenberg-Marquardt algorithm ". Department of signals and systems, Chalmers University of Technology 41296 Gothenburg, Sweden, ngia@ae.chalmers.se
7. J. Dunlop, and M.J. Al-Kind; "Application of ANC to diver voice communications", proceeding ICASSP-87, pp40.6.4, Dallas, Texas, USA, 1987.

Table (1): BP algorithm

1. Initialize (weights and Biases) (i.e Wij and bj)
2. For each training pair Do (3-9) unit performance criterion.
3. Sums the weighted input and apply activation function to compute output
   Hoj = bj + $\Sigma i =_1$ XiWij ; Wij is the weight which connects the j th node in the layer L-1 to the I th node in the layer L ; b is the bias term ; x is the input training vector ; hj are the nodes in the hidden layer
   Where: hj = f (hoj)
4. Sums the weighted output and apply activation function to compute output.
   Yok = bo_k + $\Sigma i =_1$ WjK ; k are the number of iterations
   Where yk = f (yok) ; f( ) is the sigmoid non-linearity function .
5. Compute Back propagation error.
   $\delta k = (yd_k - y_k)$ f (yok) ; δ is the portion of error correction.
   ; d is the desired response
6. Calculates weight and bias correction term.
   $\Delta Wik = \mu\ \delta khk$; μ is the learning rate
   Where $\Delta bik = \mu\ \delta k$
7. Sums the delta input for each hidden unit and calculates error term
   $\delta ok = \Sigma\delta kWlk$
   Where $\delta j = \delta ojf'$ (hok)
8. Calculate weight and bias correction term
   $\Delta Wij = \mu\delta jxi$; ΔW is the correction matrix of weights
   Where $\Delta bij = \mu\delta j$
   Update biases and weights
   Wjk (new) = Wjk (old) + Δwjk
   Bjk (new) = bjk (old) + Δbjk

10. END

Table (2) LM algorithm

Initialize network (weights and Biases)
For each training pair Do (3-7) unitl performance criterion.
Sums weighted input and apply activation function to compute output signal:
Hoi = Σi = 1 wijxi + bi
Where
hi = f (hoj)
Compute output of the network
yy = bp + Σi = 1 wpi hi
Where y = f (yy)
Calculate error term.
δ = y - yd
Calculate correction term
wb = [w1 b1 w2 b2 …… wpb p] ; p are the number of training patterns
Δ wb = (J^T.J + ηI)^ - 1. (J^T . δ)
Update weights and biases
wij (new) = wij (old) + Δwb
8. END

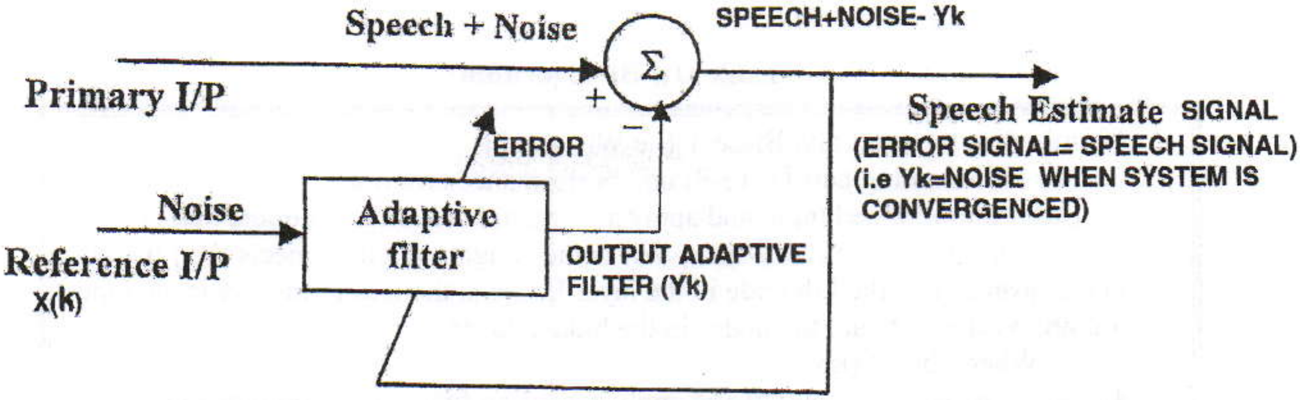


Fig (1) Adaptive Noise Cancellation "ANC" Concept

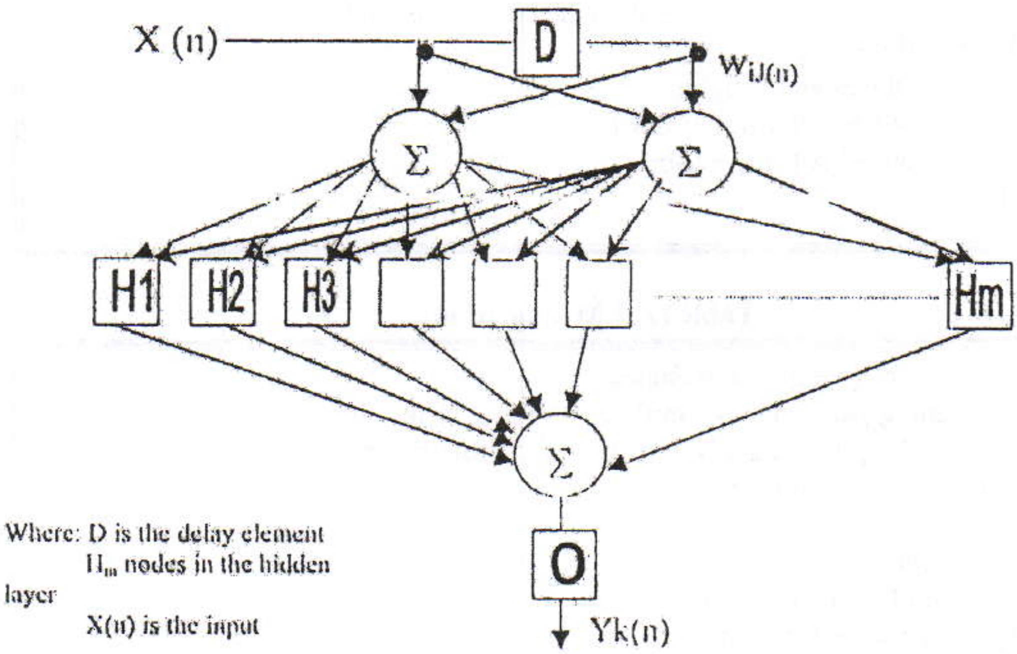


Fig (2) Three layers feed forward perceptron network

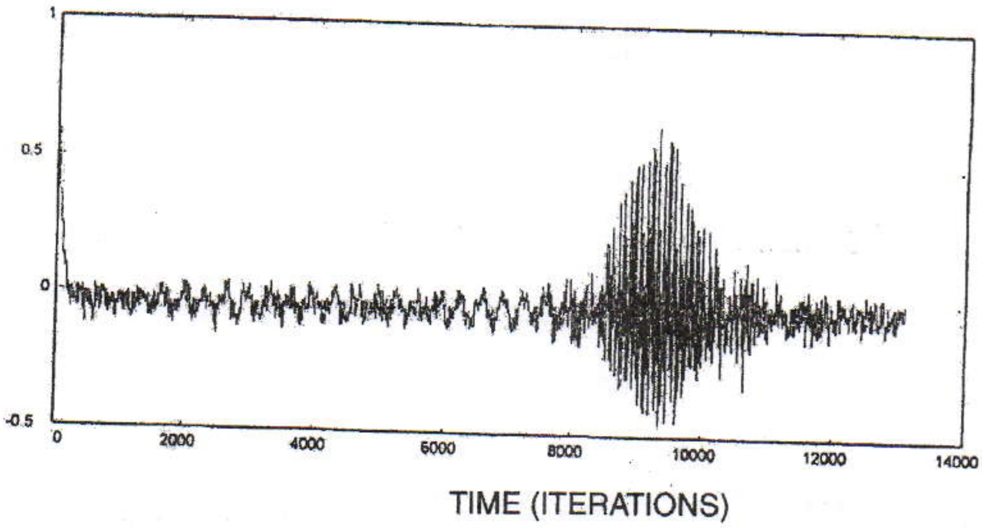


Fig (3) Information signal

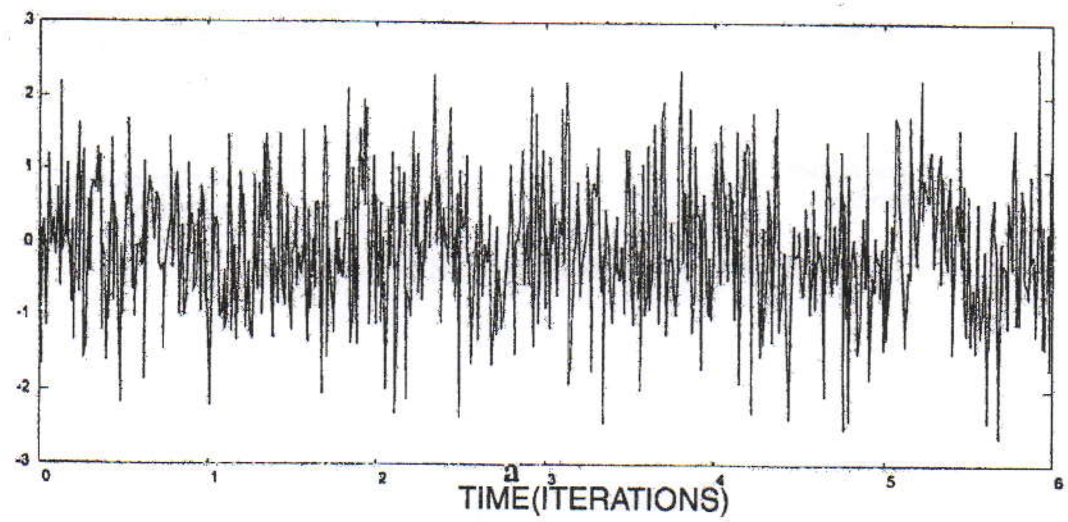
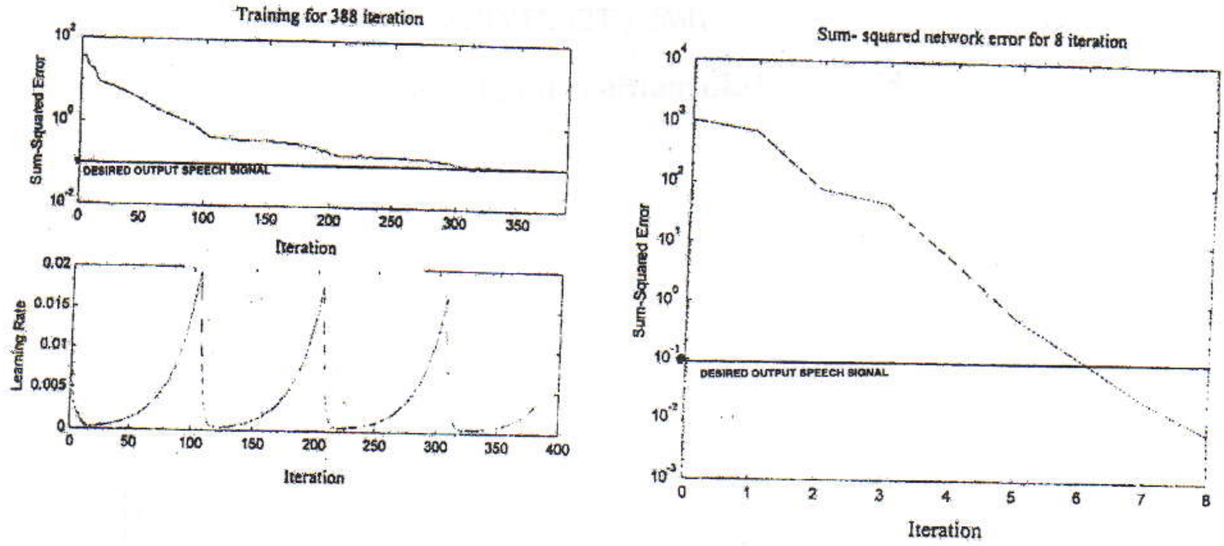


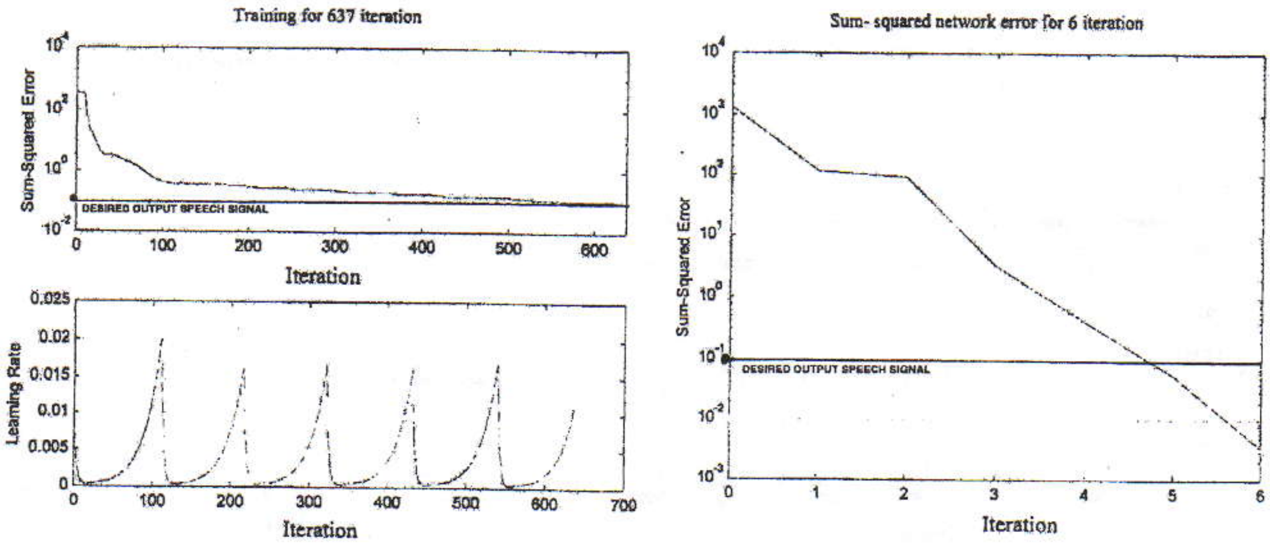
Fig (4) White Gaussian noise



Back- propagation
algorithm

Levenberg- marquardt
algorithm

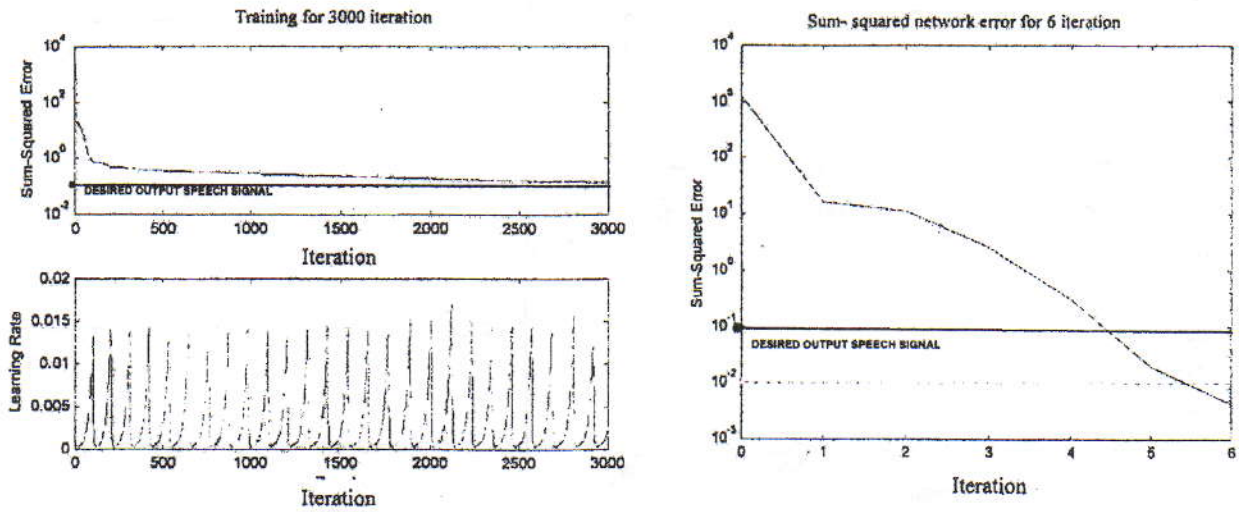
Fig(5) MSE for neural network of 10 nodes in the hidden layer



Back- propagation algorithm

Levenberg- marquardt algorithm

Fig (6) MSE for neural network of 30 nodes in the hidden layer



Back- propagation
algorithm

Levenberg- marquardt
algorithm

Fig (7) MSE for neural network of 40 nodes in the hidden layer