Equalization of Supervised Data Trained RBFNN using MSFLA

Sunita Panda, Padma Charan Sahu

Abstract: In order to avoid the channel distortion in signal processing recently, RBFNN based equalizers is mentioned. Hit and trail method is the main provocation problem for design of RBFNN Equalizer. Here the initiation is start with use of the population based optimization algorithm trained RBFNN equalizer, such as Shuffled Frog-Leaping Algorithm as well as its modified forms. The observation is made on the basis of its performance as compared to the other equalizers.

Keywords: RBFNN, Equalization Technique, SFLA.

I. INTRODUCTION

In the recent era research on channel equalization gives priority on different types of evolutionary algorithms. The description of channel equalization using different different types Artificial Neural Network and fuzzy systems as in [1]-[2]. The channel equalization techniques represent use of neural networks [3, 4] because of its limitations. As Artificial Neural Networks provides lengthy, not easy to calculate the optimum values so that a new technology used such as RBFNN provides lesser complexity and also provide better performance as compared to ANN. So Radial Basis Functional Neural Network in channel equalization is active area for the research work.

The major problems of RBFNN are that setting of parameters such as centers and widths. By using classical methods the parameter of RBFNN was determined in [5]. To overcome the drawback of existing algorithm such as Particle Swarm Optimization Algorithm and Genetic Algorithm, SFLA is used. SFLA provides simple in steps, less parameter used and easy realization. To overcome the drawback of Artificial Neural Network, Shuffled Frog Leaping Algorithm used to train RBFNN based equalizer.

II. SHUFFLED FROG LEAPING ALGORITHM

This Algorithm requires few parameters to optimize. In this technique the individual frog shows a solution to the problem. By taking in to number of frog the population is constituted, which is again categorized into different subunits, sub-memeplexes. The flow chart and the basic steps of this algorithm shown.

A. Basic steps of SFLA
1. Create an initial population randomly.
2. Fragment the individual population in to memeplexes.
3. Within individual constructed memeplexes make sure for causing memetic evolution, which may improve the quality of individual memes and also increase performance.
   The step size is
   
   \[ S = r \left( X_b - X_w \right) \]  
   \[ X_w' = X_w + S, \quad \left( S < S_{\text{max}} \right) \]

Where s be the step size, r is the random variable whose value
lies between (0,1). \( X_b \) be the best fitness of the frog. The worst fitness of the frog is \( X_w \).

From equation – (1) we observed that the step size is not sufficient to for updating the solution, therefore to increase the step size we can go for Modified form of Shuffled Frog Leaping Algorithm (MSFLA).

III. MODIFIED FORM OF SFLA

As the step size is not so large for finding the updated solution, a Mutation operator is multiplied for improving the convergence speed. The total population size is \( N \). Therefore the Eq-1 is rewritten as

\[
S = w_i \ast (X_b - X_w)r \quad i= 1,2,3,4,5, \ldots, N
\]

\[ w_i \]

is defined as

\[
w_i = w_j + (w_h - w_j) \ast X_i
\]

and is known as compressibility factor.

For increase the speed Zhang et.al used AFSA i.e Artificial Fish Swarm Algorithm for exchange of information globally, and kavousifard shows a rearrangement for updating the worst one by a new vector

IV. PROPOSED TRAINING METHOD

The basic training for RBFNN is that to determine Radial Basis Functions and centres etc. The training method is outlined below. i.e:

1. Introduce the number of memes. Every memes relating to the systems. Begin the iteration.
2. Individual meme is decoded to Artificial Neural Network, determine the weights with the help of pseudo–inverse method. And determine fitness.
3. Position to be refreshed using Shuffled Frog Leaping Algorithm.
4. Move to upcoming iteration;
5. Move to step ii till extend to utmost limit.

V. SHUFFLED FROG LEAPING ALGORITHM RADIAL BASIS FUNCTIONAL BASED EQUALIZER

While modeling the channel, The concern system uses output channel parameter to eradicate the complex process, which is exist in both channel (linear and Non linear). The prime work of this paper is to find the channel output states. This may cause trouble for the process of equalization.

For Shuffled Frog Leaping Algorithm Radial Basis Functional Based Equalizer \( N \) be the Initial population (individual memes).

VI. SIMULATION OUTCOMES

In order to find outcomes of the suggested Shuffled Frog Leaping Algorithm Radial Basis Functional Based Equalizer, Comparison is made between GA based Functional NN and PSO based Functional NN and the concerned Equalizer. The simulation parameters are shown in table 1

By using a distorted channel having the transfer function

\[
H(z) = 0.26 + 0.93z^{-1} + 0.26z^{-2}
\]

(5)

Equation (6) shows the nonlinearity which is introduced in the channel.

\[
y(n) = \tanh \left[ x(n) \right]
\]

(6)
From the above graph, suggested SRBF-J represents excellent outcomes compared to others. SRBF-J based equalizer requires 250 iteration to achieve convergence as shown in the figure where as the others are fails at this point.

Likewise graph iv shows the BER comparison among the equalizers and shows that Shuffled Frog Leaping Algorithm Radial Basis Functional Based Equalizer performs better than others. Finally the overall performance of SRBF- J is better in all aspect i.e. SNR and Bit Error Rate as compared to other based equalizer

**VII. CONCLUSION**

Here the observation is on RBFNN training of Modified form using Shuffled Frog Leaping Algorithm. Here the major contribution is comparison among the equalizers by considered to their performance (SNR and BER) and is observed that Shuffled Frog Leaping Algorithm Radial Basis Functional Based Equalizer performs well as compared to other equalizers.

**REFERENCES**