

# Neural network blind equalization algorithm Based on Feed Forward Neural Network

Huiqing Zhao

Xinhua College of Sun Yat-sen University  
Department of Information Science  
Guangzhou, China 510520  
Email: 461539139@qq.com

**Abstract**—Existing neural network algorithms have the problems of slow convergence and low accuracy. In response to this phenomenon, this paper presents a neural network blind equalization algorithm based on feed-forward neural network. And we proposed feed-forward neural network blind equalization algorithm by research of traditional neural network blind equalization algorithm. And it is using a feed-forward neural network of the hidden layer to approximate the objective function. At last, we by combining the cost functions of feed-forward network to correct the acquired information. Experimental results show that the experimental results basically consistent with the expected results. By comparison with other algorithms, this algorithm has better convergence and accuracy.

**Keywords**—Neural network; Blind equalization algorithm; feed-forward; Cost functions Convergence

## I. INTRODUCTION

In digital communication systems, digital signal crosstalk the generated between codes in the process of transmission, because the channel factors of multipath propagation and fading[1-2]. Therefore, People using the channel equalization algorithm to eliminate crosstalk between codes. Through the channel to compensate, so that the information received by the receiver to properly reload [3-4]. Traditional equalization algorithm is performed by a transmission side transmits a known training sequence to get the correct filter coefficients. And it is using these filters coefficients to changing the filter characteristics of the equalizer. At last, we can track changes in channel to achieve a better balance effect [5-6]. The blind equalization algorithms are skipping the training signal of the known. Instead, it is through the use of the receiver signal to construct cost functions. It does this by making the cost function is minimized to get the right filter. At last, it can achieve a balanced channel characteristic. Therefore, the paper chose blind equalization algorithms for data processing in the neural network.

So this paper is improved of the traditional neural network algorithm. And it is redefined the correlation coefficient of the algorithm. And the use of thresholds to control the iterative updates method of the equalizer weights. When the correlation coefficient greater than the set threshold, represents the signal is strong correlation. So we should prevent the update stagnation of equalizer weights. Therefore, this paper is

combined with the feed-forward neural network to get the best data.

## II. BLIND EQUALIZATION ALGORITHM

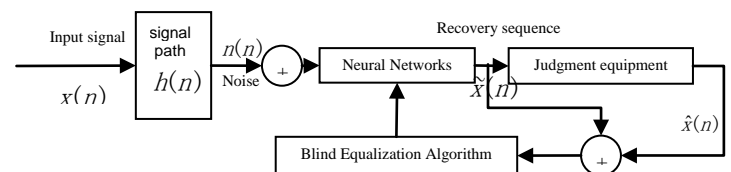


Figure 1. Schematics of the blind equalization algorithm

By observing Fig.1, we can see that system principles of blind equalization algorithm. According to the channel transmission works get the formula as follows.

$$y(n) = h(n) * x(n) + n(n)$$

There in  $x(n)$  is input signal of channel.  $h(n)$  is the impulse response of the unknown channel model.  $n(n)$  is the mean of Gaussian noise.

In the CMA blind equalization algorithm, blind equalizer is transversal filters. Set It is the weight coefficient of the transversal filter.

$$\text{We can get that: } \tilde{x}(n) = \varpi(n) * y(n)$$

According to the statistical properties of higher-order signal, we can get the cost function as follows.

$$J_d = \frac{1}{2} [ | \tilde{x}(n) |^2 - R_{cl} ]^2$$

$$\text{Therein } R_{cl} = \frac{E(| \tilde{x}(n) |^4)}{E(| \tilde{x}(n) |^2)}$$

$$\text{Get } J_d = \frac{1}{2} [ | y(n) * \varpi(n) |^2 - R_{cl} ]^2$$

Solutions for minimizing the cost function are  $\varpi(n) = \arg \min J_d$ , This is a typical problem of unconstrained optimization. By observing the cost function, we can see that  $y(n)$  and  $\varpi(n)$  can determine the value of the cost function. Therefore, we are use of "steepest gradient descent" iterative calculation method.

That the weights of the adaptive algorithm is described as:

$$\begin{aligned}\varpi_{n+1}(n) &= \varpi_n(n) - \mu \frac{\partial J_d}{\partial \varpi_n(n)} \\ \frac{\partial J_d}{\partial \varpi_n(n)} &= (|y(n) * \varpi(n)|^2 - R_{cl}) \times \frac{\partial (|y(n) * \varpi(n)|^2 - R_{cl})}{\partial \varpi_n(n)} \\ &= (|y(n) * \varpi(n)|^2 - R_{cl}) \times \frac{\partial (y(n) * \varpi(n)) (y(n) * \varpi(n))^T}{\partial \varpi_n(n)} \\ &= (|y(n) * \varpi(n)|^2 - R_{cl}) \times y(n) (\varpi(n) * y(n))^T\end{aligned}$$

Therein  $\mu$  is learning step. And make the error function as:

$$e_n(n) = (|y(n) * \varpi(n)|^2 - R_{cl}) y(n)$$

Power factor adjustment based on adaptive algorithm, we can get that  $\varpi_{n+1}(n)$ .

$$\varpi_{n+1}(n) = \varpi_n(n) - \mu e_n(n) (y(n) * \varpi(n))^T$$

But in actual communication conditions, strictly linear channel does not exist. At this point requires nonlinear blind equalizer to solve the equalization problem of nonlinear channel blind. Neural network as a dynamic nonlinear system, it have a very distinct advantage in the Blind channel equalization.

### III. ARTICLE ALGORITHM

#### A. Blind equalization algorithm base on Feed-forward neural network

Through the study of the neural network can be found using feed-forward neural network can approximate any continuous function in hidden layer. Here it is analyzed as follows [7].

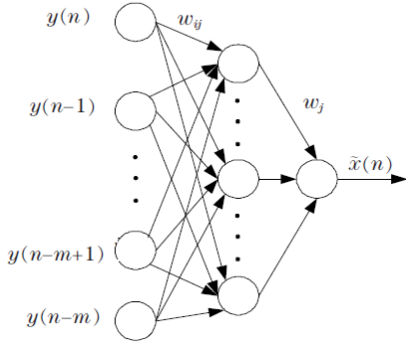


Figure 2. Architecture diagram of the feed-forward neural network

Therein  $\varpi_{ij}(n)$ ,  $i = 1, 2, \dots, m$ ;  $j = 1, 2, \dots, n$  is connect weight value of the input layer and hidden layer unit.  $\varpi_j(n)$ ,  $j = 1, 2, \dots, n$  is connection weights value of hidden layer and output layer unit. Hypothesis the enter the hidden layer units is  $u_j(n)$ , Hidden layer unit output is  $v_j(n)$ , Input of the output layer unit is  $I(n)$ , Equation of state for the the network as follows.

$$\begin{aligned}u_j(n) &= \sum_{i=1}^m \varpi_{ij}(n) \tilde{x}(n-i) \\ v_j(n) &= f[u_j(n)]\end{aligned}$$

$$I(n) = \sum_{j=1}^n \varpi_j(n) v_j(n)$$

$$\tilde{x}(n) = f[I(n)]$$

Therein  $f(x)$  is the transfer function of neural networks, here select the transfer function of the neural network as:

$$f(x) = \lambda \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (1)$$

In order to guarantee that the monotonic of transfer function, make it guide function is always greater than zero. This requires adjustment parameter  $\lambda$  is greater than zero. The study found that different signal amplitude requires different parameters. When the amplitude of the signal is large, we should use the larger  $\lambda$ . In contrast, we should use the smaller  $\lambda$ . Through the study, we can found that hidden layer and output layer input values can response the signal amplitude response transform. Therefore, the functional of Threshold  $\lambda$  as follows.

$$\lambda = \begin{cases} 1 & I(n) \geq u_j(n) \\ \exp(u_j(n) + I(n)) / T & I(n) < u_j(n) \end{cases}$$

Hidden layer unit  $u_j(n)$ , output layer unit  $I(n)$ ,  $T$  is a temperature.

$\varpi'(n)$  equivalent weights of neural networks. According to constant modulus cost function, we can get cost

$$\text{function: } J_{d1} = \frac{1}{2} [| \tilde{x}(n) |^2 - R_{cl}]^2 \quad (2)$$

And then, using BP neural network algorithm down generation algorithm is as follows.

$$\varpi(n+1) = \varpi(n) - \mu \frac{\partial J_{d1}}{\partial \varpi(n)} \quad (3)$$

$$\frac{\partial J_{d1}}{\partial \varpi(n)} = 2[\tilde{x}^2(n) - R_{cl}] \tilde{x}(n) \frac{\partial J_{d1}}{\partial \varpi(n)} \quad (4)$$

$$\text{For the output layer: } \frac{\partial J_{d1}}{\partial \varpi(n)} = f'[I(n)] v_j(n) \quad (5)$$

Combined these functions (4) (5) to substitution of the function (3), we can get that :

$$\varpi(n+1) = \varpi_j(n) + \mu H(n) v_j(n)$$

Therein  $\mu$  is a step iterative.

$$H(n) = -2[\tilde{x}^2(n) - R_{cl}] \tilde{x}(n) f'(I(n))$$

$$\text{For the hidden layer: } \frac{\partial \tilde{x}(n)}{\partial \varpi_{ij}(n)} = f'[I(n)] \frac{\partial I(n)}{\partial \varpi_{ij}(n)} \quad (6)$$

$$\text{Therein: } \frac{\partial I(n)}{\partial \varpi_{ij}(n)} = \varpi_j(n) f'[u_j(n)] y(n-i) \quad (7)$$

Combined these functions (6) (7) to substitution of the function (3), we can get that :

$$\varpi_{ij}(n+1) = \varpi_{ij}(n) + \mu H_j(n) y(n-i)$$

Therein  $H_j(n) = f'[u_j(n)] \varpi_j(n) H(n)$

In the cost function (1), According to the above iterative algorithm can be implemented in the neural network blind equalization. However, due to the non-convexity of the cost function, to avoid falling into local minima. Correction process as follows.

#### B. Define the iterative step factor

Through the study we can found that Iterative step factor  $\mu$  cannot be ignored in the convergence process [8-9]. However, these step factor  $\mu$  are fixed in most of the existing algorithms. Thus it cannot be applauded convergence result. In the algorithm, we can get different effects in different values of Iterative step factor  $\mu$ . If we using large step factor, then the adjustment is large in every tap factor. That can make the convergence of the algorithm is enhanced. But when the tap coefficients of the equalizer approach the optimum value, tap coefficients will not be able to further converge. It would be shaking back and forth within a larger range at the optimum value. It will be result a large error in the remaining. In contrast, if we use a smaller iteration step factor, then the adjustment is small in every tap factor. That can make the convergence become slow. But when the tap coefficients of the equalizer approach the optimum value, the tap coefficients shake is small. As such, steady-state error will be small. But the amount of calculation will be larger.

Therefore, in order to solve this problem, this paper proposes a may change iterative step algorithm. We are use of variable step instead of a fixed step size. We are using it to optimize algorithms. Iterative formula of tap coefficients is shown below.

The iterative formula of output layer:

$$\varpi(n+1) = \varpi_j(n) + \mu H(n) v_j(n) \quad (8)$$

Therein  $H(n) = -2[\tilde{x}^2(n) - R_{ch}]\tilde{x}(n)f'(I(n))$

Weights iterative formula of hidden layer:

$$\varpi_{ij}(n+1) = \varpi_{ij}(n) + \mu H_j(n) y(n-i) \quad (9)$$

Therein  $H_j(n) = f'[u_j(n)]\varpi_j(n)H(n)$

According the iterative formula of hidden layer and output layer, this paper is defined the iterative step factor  $\mu$ .

$$\begin{aligned} \mu &= \beta\{1 - \exp[-\alpha \cdot MSE]\} \\ &= \beta\{1 - \exp[-\alpha \cdot E(e^2(n))]\} \end{aligned} \quad (10)$$

$$e(n) = \hat{x}(n) - \tilde{x}(n)$$

Calculation it according to output signals  $\tilde{x}(n)$  of blind equalizer and Input signal  $\hat{x}(n)$  of judgment.

Therein  $\alpha$  is undetermined parameter. And it can control the speed of the step change.  $\beta$  is scaling factor. It can control the factor ranges of the iterative step. And it is satisfy that  $\beta \leq 2 / 3tr(R)$ .

Therein  $R$  is the autocorrelation matrix of the equalizer input signal.  $tr(R)$  is the trace of  $R$ .

#### C. Correction processing

Consider for the same communication channel, If the channel is minimum phase, that represent the channel can be guaranteed a reversible condition. In this case, cost functions  $J_d$  and  $J_{d1}$  have the same global minimum.

$$x(n) = J_d \cdot \tilde{x}_1(n) + J_{d1} \cdot \tilde{x}_2(n) \quad (11)$$

Therein, enter the results  $\tilde{x}_1(n)$  obtained by the transversal filter. The output value as the input of neural network input layer  $\tilde{x}_2(n)$ . cost functions satisfy of the expression  $J_d + J_{d1} = 1$ .

This paper presents a neural network under a dynamic correction, combination of transversal filter be redesigned.

#### IV. SIMULATION ALGORITHM

This paper is using the simulation probability of a binary sequence. At the same time, In order to make the simulation carried out smoothly, we have using the MATLAB to be simulation. And the use of the telephone channel and common channel as experimental. Then we using article algorithm to compare with the literature algorithm[10].

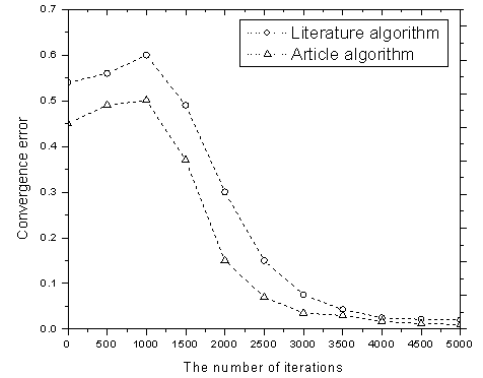


Figure 3. Convergence curve under ordinary channels

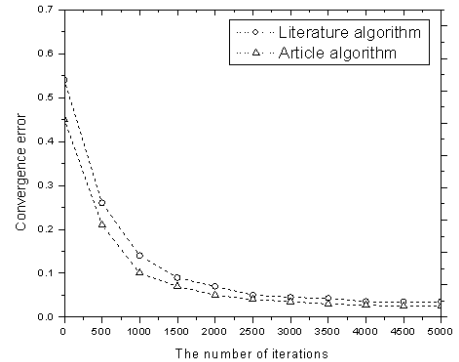


Figure 4. Convergence curve under telephone channel

By observing Fig.3-4, we can found that this algorithm is superior to literature algorithm.

Under ordinary channels, Convergence curve first increased up, but at the iteration value about 1000, it was reduce. But under telephone channel, Convergence curve diminishing from high to the low, until stable. The value of the proposed algorithm is always smaller than the literature algorithm.

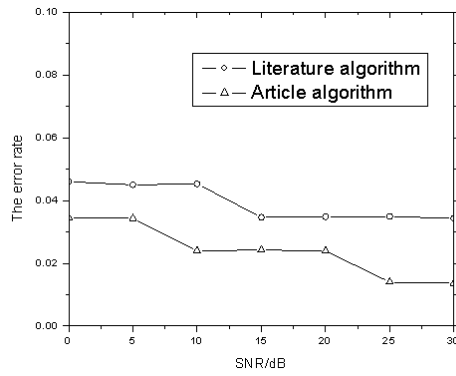


Figure 5. Error rate under ordinary channels

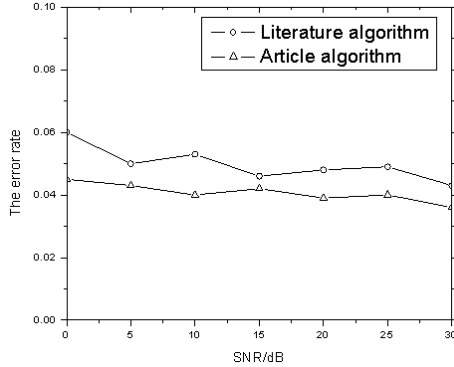


Figure 6. Error rate under telephone channel

At the same time, observing Fig.5-6 we can see that the error rate of article algorithm is minimal, too. And we can see that ordinary channels can get a better resulting value with telephone channel. And then, The error rate of article algorithm is the best.

## V. CONCLUSION

Proposed blind equalization algorithm has been modified dynamically. By combining the characteristics of blind equalization algorithm, we make it fusion with the traditional

neural network algorithm. And by improving its step and parameters, to make it get the better convergence effect. At last, it can correction processing of the output value. Experimental results show that This algorithm in terms of convergence error and speed are better other algorithms.

## REFERENCES

- [1] Khuhawar FY, Unar MA, Mahoto NA, Shazia A. Blind Channel Equalization Using Elman Network[J]. Sindh University Research Journal -Science Series. 2014,46(3): 335-343.
- [2] Xian Y, Yang L, Peng DZ, Xie SL. A second-order blind equalization method robust to ill-conditioned SIMO FIR channels [J]. DIGITAL SIGNAL PROCESSING. 2014,32: 57-66.
- [3] Chang YX, Hu GJ, Yan L, Zhao L, Li L. Mode demultiplexing based on multimodulus blind equalization algorithm [J]. OPTICS COMMUNICATIONS. 2014,324: 311-317.
- [4] Chan JC, Ma H, Saha TK. Automatic Blind Equalization and Thresholding for Partial Discharge Measurement in Power Transformer[J]. IEEE TRANSACTIONS ON POWER DELIVERY. 2014,29(4): 1927-1938.
- [5] Yang DL, Ren YB, Zhang J, Chen ZQ, Chen DH. A Variable Step-size Partial Update Blind Equalization Algorithm [J]. Telecommunication Engineering. 2014,54(7): 869-875.
- [6] Xiao Y, Yin FL. Blind equalization based on RLS algorithm using adaptive forgetting factor for underwater acoustic channel [J]. CHINA OCEAN ENGINEERING. 2014,28(3): 401-408.
- [7] Sun YS, Zhang LY, Zhang J, Shi LJ. Neural Network Blind Equalization Algorithm Applied in Medical CT Image Restoration [J]. Mathematical Problems in Engineering. 2013,10: 743546 (10 pp.)
- [8] Guo YC, Whang LH. A Hybrid Wavelet Neural Network Blind Equalization Algorithm Based on Fuzzy Controlling [J]. Acta Electronica Sinica. 2011,39(4): 975-980.
- [9] Han SH, Park Sd, Pedrycz W. Conditional fuzzy clustering for blind channel equalization [J]. APPLIED SOFT COMPUTING. 2011,11(2): 2777-2786.
- [10] Gao M, Guo YC, Liu ZX, Zhao XQ. Blind Neural Network Equalization Algorithm Based on Orthogonal Wavelet Transform Fractionally Spaced Equalizer [J]. Journal of Data Acquisition & Processing. 2011,26(2): 128-134.