# A Novel Learning Algorithm of Back-propagation Neural Network

Bing Gong
School of Computer Science and Technology
Heilongjiang University
Harbin, China
e-mail:Gongbing80@163.com

*Abstract*—**Standard neural network based on back-propagation learning algorithm has some faults, such as low learning rate, instability, and long learning time. In this paper, we introduce trust-field method and bring forward a new learning factor, meanwhile we adopt Quasic-Newton algorithm to replace gradient descent algorithm. Three algorithms are utilized in the novel back-propagation neural network. Thus the neural network avoids the local minimum problem, improves the stability and reduces the training time and test time of learning and testing. Two concrete examples show the feasibility and validity of the new neural network.**

***Keywords-Back-propagation learning algorithm (BPLA); quasic-Newton algorithm (QNA); trust-field method (TFM); self-adaptive learning factor***

## I. INTRODUCTION

Back-propagation neural network (BPNN) is one of mostly used neural networks and has been applied to many fields, such as: pattern classification, pattern recognition, self-adaptive control system, diagnosis of medical aspects or machine faults [1-4]. Traditional BPNN adopts gradient descent algorithm (GDA) which results in some problems of local minimum in learning, bad convergence, instability and long training time. Due to BPNN's practicability, many optimal methods, such as momentum factor, Levenberg-Marquardt (LM), QNA, Kalman Filter and its improved algorithms, are used to improve the capacity of standard BPNN [5-7].

In the research of BPNN, includes several difficult points, as follows: the choice of learning factor, the numbers of hidden layers, the number of hidden layer's units, and the error learning and updating algorithm. Aiming at these problems, the new neural network based on improved back-propagation learning algorithm (NBPNN) is proposed to overcome these faults. For the choice of learning factor, the present emphasis concentrates on how to choose appropriate learning rate depending on expertise and experience. Here we utilize the real-time output and experience of expertise and experience, propose a new adaptive-learning factor (ALF). Aiming at the traditional learning algorithm, GDA, use QNA to replace GDA, meanwhile use TFM can avoid the local minimum and improve the efficiency.

This paper includes three two keys, ALF and TFM, meanwhile depends on QNA learning algorithm. TFM and QNA can not only solve the local minimum problem but also reduce the training time. ALF can make NNN-IBPLA have better accuracy compared with standard BPNN.

The rest of this paper is organized as follows. In Section 2, we recall some basic concepts. In Section 3, we present the structure of the novel BPNN structure. In Section 4, we utilize two examples to validate the novel BPNN. At last a conclusion is obtained in Section 5.

## II. PRELIMINARY

In this Section, we recall some basic concepts, i.e. standard BP neural network, TFM algorithm and the BPNN structure based on QNA algorithm.

### A. Standard BP Neural Network

Standard BPNN belongs to one of classic feed-forward neural networks, and its learning algorithm is GDA. Presently LM algorithm has been applied in BPNN generally and some optimal algorithms have become the research focus. In this section, the structure of standard BP neural network is as follows:

Assume the NN include three layers, input layer, a hidden layer and output layer. The input vector is denoted by $x=(x_1,\cdots,x_n)^T$, the output vector of hidden layer is $y=(y_1,\cdots,y_m)^T$, the actual output vector is $o=(o_1,\cdots,o_l)^T$, the expected output vector is $d=(d_1,\cdots,d_l)^T$. The weight matrix between input layer and hidden layer is denoted by $v=(v_1,\cdots,v_m)^T$, the weight matrix between hidden layer and output layer is $w=(w_1,\cdots,w_l)^T$. The activation function between different layers adopts sigmoid function $f(x)=\frac{1}{1+e^x}$. The objective of BPNN is to make the input-output problem be an optimal with no constraint. For the output of NN, we defines a energy function to be a optimal problem as

$$\min_w \; E(w)=\sum_{i=1}^{l}\frac{1}{2}\left(d_i-o_i\right)^2 \tag{1}$$

The objective is to obtain the minimum of the energy function, BPNN includes feed-forward propagation and back propagation steps. The feed-forward propagation is a nonlinear process, the kernel of the BP neural network is back propagation.

If the error between actual output and expected output more than the set error, then we update the weight matrix by back propagation learning. In standard BPNN, it adopts minus gradient to adjust weight matrix.

$$\Delta v_{ij}=-\eta\frac{\partial E}{\partial v_{ij}},i=0,1,\cdots,n;\;\; j=1,\cdots,m \tag{2}$$

978-0-7695-3728-3/09 $25.00 © 2009 IEEE
DOI 10.1109/CASE.2009.146

$$\Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}}, j = 0,1,\cdots,m;\ \ k = 1,\cdots,l \quad (3)$$

The new weight matrix is updated as

$$w_{jk} = w_{jk} + \eta \Delta w_{jk} \quad (4)$$

This is an iterative process until the error obtains the concessional threshold, obviously it can not obtain the expected error due to adopt the gradient descent algorithms.

*B. Principal of TFM*

The most important advantage of TFM is its global convergence property. Its principal is as follows: assume $k$th interactive step be finished, for finding feasible $d_k$, such that $x_{k+1} = x_k + d_k$ is the apex in $\{x_k + d, \|d\| \le \Delta_k\}$. Where $\Delta_k$ has a constraint as: in the sphere $\{x_k + d, \|d\| \le \Delta_k\}$, the quadratic function model approximates to the objective function properly. Generally speaking, TFM is an interactive process in different sphere, in every step, it seeks out the present optimal direction.

*C. BPNN based on QNA Algorithm*

Compared with GDA, QNA can solve the Hessen matrix problem well. In BPNN structure, assume that the NN has done he *(k-1)*th iterative and the *k*th feed-forward propagation and obtain the energy error function $E(w)$. Then we do transform of $E(w)$ and obtain

$$\begin{aligned} E(w) &= E(w_k) + g(w_k)(w - w_k) \\ &+ \frac{1}{2}(w - w_k)G(w_k)(w - w_k) \\ &+ O((w - w_k)^2) \end{aligned} \quad (5)$$

Let $E'(w) = 0$, then

$$\begin{aligned} g(w_k) &= -(g_k^T g_k + \Delta_k)(w_{k+1} - w_k) \\ &= -G_k P_{k+1} \end{aligned} \quad (6)$$

$$w_{k+1} - w_k = P_{k+1} = -G_k^{-1} g(w_k) \quad (7)$$

We use QNA to update weight matrix as

$$P_{k+1} = -H_k g_k \quad (8)$$

$$w_{k+1} = w_k + P_{k+1} \quad (9)$$

Then we obtain $w_{k+1} = w_k - H_k g_k$

The concrete process of BPNN based on QNA as follows:

*a) Input training samples.*

*b) Do* $E'(w) = 0$ *and obtain correlative parameters*

*c) Adjust* $H_{k-1}$ *and* $H_k$ *by Step 2.*

*d) Obtain the descent orientation* $P_{k+1} = -H_k g_k$ *.*

*e) Have a search and obtain learning factor* $\eta$*, compute* $w_{k+1} = w_k + \eta P_{k+1}$*, repeat the whole process.*

## III. THE NOVEL BPNN STRUCTURE

In before-mentioned part, we give an introduction to standard BPNN, TFM and QNA. In this section, the emphasis is the adaptive-learning factor (ALF) and the whole structure of NBPNN.

*A. ALF*

After the *(k-1)*th iterative and the *k*th feed-forward propagation, the output vector is $o = (o_1^k, \cdots, o_l^k)^T$, the expected output vector is $d = (d_1, \cdots, d_l)^T$. We define the learning factor $\eta$ as

$$\eta = \frac{\sigma}{\exp\left(-\frac{1}{l}\sum_{i=1}^{l}\left(o_i^k - d_i\right)^2\right)} \quad (10)$$

Where $\sigma\left(0 \le \sigma \le 1\right)$, form the style $\eta$, if the error is great then $\eta$ will obtain a corresponding great updating value which accords with the actual conditions.

*B. The Whole Structure of The Novel BPNN*

In the whole structure of the novel BPNN (NBPNN), the learning process will be a combination of QNA and TFM, the learning factor will adopts the proposed algorithm, the whole process as follows:

Step 1: Initialize weight matrix, compute error function and give initialization value of $H_0 = I$, $P = -g$ and $K = 0$.

Step 2: Obtain direction $P$ by (8)

Step 3: Assume the *k*th interactive be finished, update direction $P_{k+1}$ by TFM, compute error function to obtain feasible direction $P_{k+1}$

Step 4: Adjust weight by $P_{k+1}$ and present learning factor $\eta$

Step 5: Compute the present error compared with expected error, if the error conditions satisfy the objective then end the whole process, else do the next iterative process.

## IV. SIMULATION AND ANALYSIS

*A. Experiment 1*

2-dimensional XOR problem, $1 \oplus 1 = 0$, $1 \oplus 0 = 1$, $0 \oplus 1 = 1$ and $0 \oplus 0 = 0$.

Where we give same initialization value for standard BPNN, BPNN based on LM and NBPNN. All NN adopts same structure, i.e. one 2-dimensional input layer, one hidden layer with 5 units and one 1-dimensiaonal output layer, and the sigmoid activation function. In the structure, the iterative numbers do not exceed 3000, the error do not exceed 0.01, $\sigma = 0.5$. Training samples are 200, and testing samples are 200. In the 200 testing, depending on three different algorithms, i.e. BPNN based on GDA algorithm, BPNN based on LM algorithm, and the novel BPNN (NBPNN) the concrete result is as TABLE 1and TABLE 2.

TABLE I. RESULT OF 2-DIMENSIONAL XOR PROBLEM (1)

| Method | Number of Local minimum | Number of more than expected precision | Average times |
|---|---|---|---|
| GDA | 14 | 11 | 1417 |
| LM | 4 | 8 | 1726 |
| NBPNN | 0 | 1 | 431 |

TABLE II. RESULT OF 2-DIMENSIONAL XOR PROBLEM (2)

| Method | The best precision | Average precision | Eligible number |
|---|---|---|---|
| GDA | 0.0075 | 0.0373 | 148 |
| LM | 0.0016 | 0.0212 | 149 |
| NBPNN | 0.0027 | 0.0139 | 165 |

### *B. Example 2*

Simulation of 2-dimensional function $z = \frac{\sin \pi y}{2 + \sin 2\pi x}$, $-0.5 \le x \le 0.5$, $0 \le y \le 1$.

The network includes 3 layers, with 2-dimensional input vector, 6-dimensional vector of hidden layer, 1-dimensional vector of output layer and $\sigma = 0.7$. Choose training samples as

$$x(i) = -0.5 + 0.002 \times i \left(i = 1, 2, \ldots, 450\right) \tag{11}$$

$$y(i) = 0.002 \times i \left(i = 1, 2, \ldots, 450\right) \tag{12}$$

Testing samples are as

$$x(i)' = -0.5 + 0.002 \times i + 0.003\left(i = 1, 2, \ldots, 450\right) \tag{13}$$

$$y(i)' = 0.002 \times i + 0.003\left(i = 1, 2, \ldots, 450\right) \tag{14}$$

Then we can obtain Figure 1, Figure 2 and Figure 3.

According to the two examples, TABLE Ⅰ and TABLE Ⅱ, we can see in the same initialization conditions, NBPNN solves the local minimum problem completely, precision increases more than 1 times compared with BPNN based on GDA learning algorithm or LM learning algorithm, average times are about 25% of GDA or LM. From Figure 1 to Figure 3, we can see that the NBPNN has smaller error compared with two other algorithms.

In a word, the new learning algorithm of BP neural network can obtains more accurate results than some existent learning algorithms.

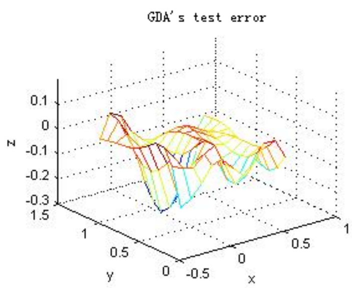


Figure 1. Error of testing samples depending on GDA algorithm

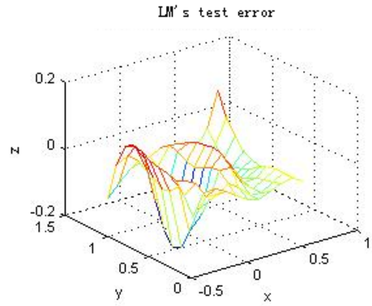


Figure 2. Error of testing samples depending on LM algorithm

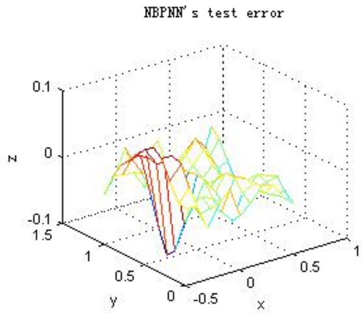


Figure 3. Error of testing samples depending on NBPNN

## V. CONCLUSION

In this paper, we propose a novel BP neural network based on the self-adaptive learning factor, TFM algorithm, and QNA learning algorithm. In a word, the major contribution includes two aspects, i.e. TFM's application combined with QNA and ALF's improvement. The former factor solves the local minimum problem and the long-time training and test, the latter factor ensure the learning precision. Simulation shows the feasibility and validity of the novel BP neural network.

## REFERENCES

[1] Maciej A. Mazurowski, Piotr A. Habas, Jacek M. Zurada, Joseph Y. Lo, Jay A. Baker, Georgia D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," Neural Networks, vol. 21, 2008, pp. 427-436.

[2] Iveta Mrázová, Dianhui Wang, "Improved generalization of neural classifiers with enforced internal representation," Neurocomputing, vol. 70, 2007, pp. 2940-2952.

[3] Xiaoping Qiu, Ning Tao, Yun Tan, Xinxing Wu, "Constructing of the risk classification model of cervical cancer by artificial neural network," Expert Systems with Applications, vol. 32, 2007, pp. 1094-1099.

[4] L. Barelli, G. Bidini, F. Mariani, M. Svanziroli, "A non-conventional quality control system to detect surface faults in mechanical front seals," Engineering Applications of Artificial Intelligence, vol. 21, 2008, pp. 1065-1072.

[5] Wang Jianmei, Jia wenzhong, "BP Neural Network Classifier Based on L-M algorithm," Journal of Wuhan University, vol. 30, 2005, pp. 928-934.

[6] Ralph Linsker, "Neural network learning of optimal Kalman prediction and control," Neural Networks, vol. 21, 2008, pp. 1328-1343.

[7] Xu Dachnan, Liu Guanghui, “A new sufficient condition for the convergence of the DFP algorithm with Wolfe line search,” Systems Science and Mathematical Sciences, vol. 9, 1996, pp. 259-269.